

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ**

Факультет Кібербезпеки, комп'ютерної та програмної інженерії  
Кафедра комп'ютеризованих систем управління

ДОПУСТИТИ ДО ЗАХИСТУ  
Завідувач кафедри

\_\_\_\_\_ Литвиненко О.Є.

«\_\_» \_\_\_\_\_ 2021 р.

**ДИПЛОМНИЙ ПРОЕКТ**  
(ПОЯСНЮВАЛЬНА ЗАПИСКА)

**ВИПУСКНИКА ОСВІТНЬО-КВАЛІФІКАЦІЙНОГО РІВНЯ**  
**"БАКАЛАВР"**

Тема: Програмний модуль опанування іноземної мови на платформі *Unity*

Виконавець: \_\_\_\_\_ Іванова М.С.

Керівник: \_\_\_\_\_ Апенько Н.В.

Нормоконтролер: \_\_\_\_\_ Тупота Є.В.

**Київ 2021**

НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ

Факультет кібербезпеки, комп'ютерної та програмної інженерії

Кафедра комп'ютеризованих систем управління

Спеціальність 123 «Комп'ютерна інженерія»  
(шифр, найменування)

Освітньо-професійна програма «Системне програмування»

Форма навчання денна

ЗАТВЕРДЖУЮ  
Завідувач кафедри

«    »      202   р.

**ЗАВДАННЯ**

**на виконання дипломної роботи (проєкту)**

Іванової Марини Сергіївни

(прізвище, ім'я, по батькові випускника в родовому відмінку)

**1. Тема дипломної роботи (проєкту):** “Програмний модуль опанування іноземної мови на платформі *Unity*”

затверджена наказом ректора від «  04  »      лютого 2021 р. № 135/ст.

**2. Термін виконання роботи (проєкту):** з   17.05.2021   до   20.06.2021  

**3. Вихідні дані до роботи (проєкту):**

  1) вимоги до роботи програмного модуля;

  2) основний функціонал програмного модуля.

**4. Зміст пояснювальної записки:**

  1) Порівняльний аналіз лінгвістичних програмних засобів;

  2) Обґрунтування підходу до розробки програмного модулю;

  3) Моделювання роботи програмного модулю;

  4) Розробка програмного модулю для вивчення іноземних мов на платформі *Unity*;

  5) Тестування програмного модулю.

**5. Перелік обов'язкового графічного (ілюстративного) матеріалу:**

  1) Діаграма класів для модулю «Vocabulary»;

  2) Діаграма класів для модулю «Grammar»;

  3) Діаграма варіантів використання;

4) Алгоритмічний опис;

5) Знімки екрану програмного модулю.

## 6. Календарний план-графік

№ п/п	Етапи виконання дипломного проекту	Термін виконання етапів	Примітка
1	Провести аналіз літератури за темою дипломного проекту та аналіз існуючих програмних додатків	17.05.21-18.05.21	
2	Зробити вибір компонентів програмного модулю	18.05.21-19.05.21	
3	Розробити структуру програмного модулю	20.05.21-21.05.21	
4	Розробка програмного модулю	24.05.21-28.05.21	
5	Провести тестування програмного модулю	31.05.21-01.06.21	
6	Написати пояснювальну записку	02.06.21-07.06.21	
7	Підготувати презентацію та графічні матеріали	08.06.21-11.06.21	

7. Дата видачі завдання « 17 » травня 2021 р.

Керівник дипломного проекту \_\_\_\_\_ Апенько Н.В.  
(підпис)

Завдання прийняв до виконання \_\_\_\_\_ Іванова М.С.  
(підпис випускника)

## РЕФЕРАТ

Пояснювальна записка до дипломного проекту “Програмний модуль опанування іноземної мови на платформі *Unity*”: 61 с., 18 рис., 1 табл., 17 літературних джерел, 2 додатки.

*EDUTAINMENT*, ВЕБ-ДОДАТОК, ІНОЗЕМНА МОВА, ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ, КОМП’ЮТЕРНА ПРОГРАМА, ПЛАТФОРМА *UNITY*.

Об’єкт дослідження – створення веб-додатку для вивчення іноземної мови, що буде мати завдання різного типу та складності.

Предмет дослідження – створення веб-додатку на платформі *Unity*.

Мета дипломного проекту – розробити програмний модуль для опанування іноземної мови в режимі онлайн використовуючи платформу *Unity*.

Методи дослідження – при вирішенні поставлених завдань було використано теоретичні відомості та практичні напрацювання в галузі створення веб-додатків на платформі *Unity*, а також інформацію про підходи до ефективного вивчення іноземних мов.

Отримані результати та їх новизна – в результаті виконання проектування було отримано програмний модуль для опанування іноземних мов, що працює в режимі онлайн, має завдання різного типу та рівня складності для опрацювання різних аспектів використання мови.

Результати досліджень можуть бути застосовані наступним чином – даний програмний модуль в своїй роботі мають змогу використовувати вчителі середньої школи для перевірки знань учнів або в якості тренажерів, батьки для розвитку своїх дітей та навіть дорослі, які прагнуть підвищити рівень володіння іноземною мовою.

Прогнози припущення щодо розвитку об’єкта дослідження – отриманий програмний продукт в майбутньому можна вдосконалити за допомогою додавання до нього інших іноземних мов (дане дослідження проводилося на основі використання англійської мови), створення особового кабінету для моніторингу успішності користувача.

## ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ, ТЕРМІНІВ .....	7
ВСТУП.....	8
РОЗДІЛ 1 ПОРІВНЯЛЬНИЙ АНАЛІЗ ЛІНГВІСТИЧНИХ ПРОГРАМНИХ ЗАСОБІВ .....	13
1.1. Загальні питання використання ІТ в освіті .....	15
1.2. Створення комп'ютерних засобів для навчання.....	17
1.3. Аналіз існуючих рішень.....	20
1.4. Висновки до розділу .....	27
РОЗДІЛ 2 ОБГРУНТУВАННЯ ПІДХОДУ ДО РОЗРОБКИ ПРОГРАМНОГО МОДУЛЮ.....	29
2.1. Платформа Unity .....	29
2.2. Особливості обудови онлайн-додатку за допомогою WebGL ..	32
2.3. IL2CPP та Emscripten.....	40
2.4. Діаграми .....	42
2.5. Висновки до розділу .....	49
РОЗДІЛ 3 ПРОЄКТУВАННЯ ТА РОЗРОБКА ПРОГРАМНОГО МОДУЛЮ .	50
3.1. Формування вимог до програмного модулю .....	50
3.2. Опис програмного модуля .....	51
3.3. Написання коду .....	51
3.4. Висновки до розділу .....	54
РОЗДІЛ 4 ТЕСТУВАННЯ ПРОГРАМНОГО МОДУЛЮ ТА ФОРМУВАННЯ ІНСТРУКЦІЙ КОРИСТУВАЧА.....	55
4.1. Функціональне тестування .....	55
4.2. Формування інструкцій для користувача.....	57
4.3. Висновки до розділу .....	58
ВИСНОВКИ.....	59
СПИСОК БІБЛІОГРАФІЧНИХ ПОСИЛАНЬ ВИКОРИСТАНИХ ДЖЕРЕЛ.....	61
ДОДАТОК А .....	63
ДОДАТОК Б.....	67



## ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ, ТЕРМІНІВ

<i>API</i>	–	<i>Application Programming Interface</i>
<i>CALL</i>	–	<i>Computer-Assisted Language Learning</i>
<i>CORS</i>	–	<i>Cross-Origin Resource Sharing</i>
<i>HTML</i>	–	<i>HyperText Markup Language</i>
<i>HTTP</i>	–	<i>HyperText Transfer Protocol</i>
<i>LLVM</i>	–	<i>Low Level Virtual Machine</i>
<i>LTS</i>	–	<i>Long Time Support</i>
<i>OpenGL</i>	–	<i>Open Graphics Library</i>
<i>SDL</i>	–	<i>Simple DirectMedia Layer</i>
<i>UI</i>	–	<i>User Interface</i>
<i>UML</i>	–	<i>Unified Modeling Language</i>
<i>UX</i>	–	<i>User Experience</i>
<i>WebGL</i>	–	<i>Web-based Graphics Library</i>
<i>WebRTC</i>	–	<i>Web Real Time Communications</i>
AM	–	англійська мова
ЗОШ	–	загальноосвітня школа
ІМ	–	іноземна мова
ІКТ	–	інформаційні комп'ютерні технології
ІТ	–	інформаційні технології
КНС	–	комп'ютерна навчальна система
КП	–	комп'ютерна програма
КТ	–	комп'ютерні технології
НП	–	навчальний процес
ОС	–	операційна система
ПЗ	–	програмний засіб
ПЗСО	–	повна загальна середня освіта
ПК	–	персональний комп'ютер
ПМ	–	програмний модуль

## ВСТУП

Вивчення ІМ наразі є актуальним питанням для більшості людей. Вивчати ІМ можна різними способами: із репетитором, на курсах, за допомогою підручників, дивлячись фільми мовою оригіналу, деякі навіть їдуть до країни, мову якої бажають вивчити, адже мовне середовище сприяє кращому та швидшому засвоєнню. Проте, навчання потребує систематичності і тому із розвитком ІТ з'явилися ПЗ, що допомагають регулярно поповнювати словниковий запас, вивчати граматику, розвивати навички читання, письма та аудіювання.

Перевага ПЗ перед підручниками у тому, що, по-перше, після виконання завдання одразу можна дізнатися результат засвоєння матеріалу, по-друге, вивчення мови відбувається в ігровій формі, що сприяє заохоченню не тільки дорослих, а також дітей та підлітків до НП, по-третє враховуючи різноманіття ПЗ, можна не виходячи із дому, дистанційно, знайти носія мови та спілкуватися із ним у форматі онлайн – переписуватися чи за допомогою відеозв'язку.

ПЗ допомагають також зекономити час. Наприклад, вивчати нові слова рекомендується використовуючи флеш-картки, тобто коли на папірці невеликого розміру з одного боку написане слово, а з іншого – переклад або для кращого ефекту пояснення тією мовою, що вивчається. Звісно, такі папірці можна зробити самостійно або купити, але разом із тим, можна завантажити додаток в телефон, записувати слова та їх дефініції і мати такі флеш-картки завжди при собі та коли зручно: в громадському транспорті або під час обідньої перерви повторювати необхідний матеріал.

Наразі, існує таке поняття як *edutainment* (від англ. *education* – навчання, *entertainment* – розвага), тобто навчання в ігровій формі. Тому, використання таких додатків допоможе вчителям та батькам в школах зацікавити дітей, а також частіше перевіряти рівень їх навчальних досягнень у вивченні ІМ. [1]

Розглянемо більш детально інтеграцію ІКТ у навчальний процес ЗОШ. КТ можна використовувати на всіх етапах навчання: для пояснення нового матеріалу,



при повторенні, закріпленні та навіть контролі. Але разом із тим, існує проблема: в багатьох школах використання інтернету та ІТ поки обмежено та майже не пов'язано із освітнім процесом.

Вивчення ІМ з використанням ІКТ та різноманітних КП відрізняється низкою переваг. Таким чином, доктор педагогічних наук, професор Є.С.Полат виокремлює наступні:

- зацікавленість учнів КТ найчастіше сприяє високій мотивації в процесі навчання;
- використовуючи КП під час навчання учні підвищують загальну, комп'ютерну та мовленнєву культуру;
- процес навчання набуває індивідуального характеру;
- об'єктивна оцінка;
- забезпечується ефективне виконання вправ.

Переваги КП:

- вільний режим навчання (самостійний вибір часу, перерв та темпу роботи);
- об'єктивна оцінка відповідей.

Серед недоліків варто виділити такі:

- складності в передачі великих об'ємів інформації;
- необхідність у висококваліфікованих спеціалістах для підтримки та оновлення програмної системи і даних.

При якісному вивченні ІМ з використанням ІКТ передбачаються такі цілі:

- фонетика: формування навичок розрізнення звуків ІМ, навичок ритміко-інтонаційної вимови;
- лексика: розширення словникового запасу, формування продуктивних лексичних навичок, що використовуються на письмі, закріплення знань за допомогою тестових та ігрових КП;
- граматики: опанування та закріплення граматичних навичок за допомогою інтерактивного тестування, вдосконалення читання та аудіювання, формування продуктивних граматичних навичок, що використовуються на

письмі;

– читання: навчання техніці читання вголос, закріплення лексичних і граматичних навичок читання, набуття навичок виділення із текстів різних видів ідеї та основної думки, навчання різним технікам аналізу тексту і контроль правильності та глибини розуміння прочитаного.

– аудіювання: формування фонетичних навичок аудіювання, контроль правильності розуміння прослуханого тексту.

– мовлення: формування лексичних, граматичних та фонетичних навичок мовлення.

Слід зазначити, що існують різні форми використання ІКТ в умовах сучасного НП:

– ІКТ застосовуються як додаткові технічні засоби при демонстрації наочних матеріалів та поясненні учбового матеріалу та/або для самостійної роботи учнів під керівництвом викладача;

– для роботи учня в індивідуальному режимі;

– для дистанційного навчання, коли комп'ютерні засоби використовуються для забезпечення зв'язку учня та вчителя;

– для колективної творчої роботи учнів;

– для виконання завдань учнями за одним комп'ютером при вдосконаленні мовленнєвих навичок.

В ХХІ столітті існує таке поняття як комп'ютерна лінгводидактика. Комп'ютерна лінгводидактика – це галузь лінгводидактики, що вивчає теорію та практику в ході використання ІКТ для вивчення ІМ. Наразі вчені виділяють два етапи розвитку даної науки: традиційний та сучасний.

Перший етап – традиційний – період 60-80-х років ХХ століття. В цей час комп'ютери використовувалися з метою тренування окремих видів мовленнєвої діяльності. Також увага приділялась різним аспектам ІМ. Найбільш розповсюдженими типами програм цього періоду можна вважати тренувальні, тренувально-контролюючі, що акцентувалися на вивченні, вдосконаленні та закріпленні навичок лексики, граматики, читання та письмової діяльності.

Другий етап – сучасний – розпочався у 90-х роках минулого століття. З початком цього етапу, розвитку набувають засоби мультимедіа, телекомунікаційні технології, а також відбувається зміна технічних можливостей комп'ютера. Комплекс цих заходів дав змогу за допомогою ІКТ опрацьовувати всі аспекти використання мови. Також, важливо зазначити, що в цей період з'являється можливість реального спілкування з носіями мови не тільки в письмовій формі, але і усній. Використання комп'ютерів в цей час розвивається в двох напрямках: перший – для вивчення рідної мови, другий – для вивчення ІМ як у мовленнєвому середовищі, так і поза ним. [2]

Використання ІКТ на уроках ІМ – один із видів організації навчального процесу. В наш час на ринку представлена велика кількість програм, що дозволяє вивчати мову та вдосконалювати комунікативні навички використовуючи комп'ютерні засоби.

Актуальність теми – обумовлена попитом на КП, що можна використовувати у НП, які допомагають опрацьовувати різні аспекти використання мови, маючи при цьому градацію складності завдань.

Мета виконання дипломного проєкту – полягають у розробці ПМ для опанування ІМ в режимі онлайн використовуючи платформу *Unity*.

Основними завданнями виконання даного проєкту є дослідження процесу створення веб-додатку використовуючи платформу *Unity* і бібліотеку *Web-GL*, а також дослідження ринку КП для вивчення мови.

Об'єкт – створення веб-додатку для вивчення іноземної мови, що буде мати завдання різних типів та рівнів складності.

Предмет – створення веб-додатку на платформі *Unity*.

Методи дослідження – при вирішенні поставлених завдань було використано теоретичні відомості та практичні напрацювання в галузі створення веб-додатків на платформі *Unity*, а також інформацію про підходи до ефективного вивчення ІМ.

Наукова новизна отриманих результатів – в результаті виконання проєктування отримано ПМ для опанування ІМ, що працює в режимі онлайн, має

завдання різних типів та рівнів складності для опрацювання різних аспектів використання мови.

Практичне значення – даний ПМ в своїй роботі мають змогу використовувати вчителі середньої школи для перевірки знань учнів або в якості тренажерів батьки для розвитку своїх дітей та навіть дорослі, які прагнуть підвищити рівень володіння ІМ.

## РОЗДІЛ 1

### ПОРІВНЯЛЬНИЙ АНАЛІЗ ЛІНГВІСТИЧНИХ ПРОГРАМНИХ ЗАСОБІВ

Невід’ємною складовою отримання ПЗСО є проходження курсу ІМ. Традиційна парадигма системи вивчення ІМ поповнюється новими підходами до опанування дисципліни.

Сучасне учбове середовище називають комп’ютерно-орієнтованим, оскільки це поняття вміщує в собі використання КТ у навчанні. Наразі, дослідники акцентують увагу на тому, що основними напрямками застосування КТ на заняттях є використання мультимедійних можливостей комп’ютерних класів та освітніх ресурсів мережі Інтернет. Таким чином з’явився напрям, що синтезує використання ІКТ та вивчення мови: *CALL* – це підхід до вивчення мови таким чином, що комп’ютер виступає в якості допоміжного інструменту для представлення та вивчення матеріалу, а також оцінювання отриманих і засвоєних знань.

Останнім часом не складно помітити, що спектр функцій використання КТ при вивченні ІМ стає більш розширеним та різноманітним. Окрім базової функції трансляції інформації, ІКТ часто використовують як засіб створення інтерактивності при виконанні різного плану завдань. [3]

При вивченні ІМ етапи занять повинні бути сформовані відповідно до етапів опанування мовного матеріалу. Перший етап передбачає сприймання мовлення (проведення бесід для мотивації учнів, підготовку через подолання труднощів із граматикою та лексикою, попереднє читання або прослуховування, перевірку розуміння прочитаного та почутого). Етап аналізу мови включає в себе виділення та пояснення певних мовних феноменів, розбір лексичних і граматичних правил. На етапі закріплення виконуються вправи на окремі мовні

Кафедра КСУ				НАУ 21 20 53 000 ПЗ			
<b>Виконав</b>	Іванова М. С.			Порівняльний аналіз лінгвістичних програмних засобів	<b>Літера</b>	<b>Аркуш</b>	<b>Аркуші</b>
<b>Керівник</b>	Апенько Н. В.					12	61
<b>Консульт.</b>					СП-435 123		
<b>Норм. контр.</b>	Тупота С. В.						
<b>Зав. Каф.</b>	Литвиненко О.С.						

явища. На заключному етапі вивчений матеріал використовують в мовленні, розглядаючи різні комунікативні ситуації.

Щоб якісно дослідити питання використання КП при вивченні мови, в даному дипломному проєкті за основу будемо використовувати англійську мову.

Кількість створених КП для вивчення АМ вже перевищує кількість «звичайних» шкільних підручників, проте їх якість досі залишається незадовільною. Так відбувається тому, що всі КП базуються на основі вже готових мультимедійних комп'ютерних курсів, що пропонують вправи для закріплення матеріалів з лексики, граматики та фонетики. Матеріал даних учбових «посібників» представлений у вигляді інтерактивних вправ і часто складається із теоретичного та практичного розділів за різними аспектами, що включені в обов'язковий мінімум вивчення англійської мови. Частина такої програми може використовуватися як додатковий матеріал на уроці. В програмі може бути передбачений зворотній зв'язок при виконанні певних операцій користувачем, а саме відмічаються успіхи та, за необхідності, з'являються сповіщення-підказки. Навчальні КП рекомендовані для застосування на всіх етапах опанування мови, при формуванні та вдосконаленні лексичних, граматичних та фонетичних навичок. У зв'язку із цим повноцінне та систематичне використання навчальних КП та систем в рамках вивчення ІМ в ЗОШ не є можливим. Вчителі мають змогу використовувати тільки фрагменти програм на певних етапах навчальних занять за деякими темами, які лише частково співпадають з матеріалом, що запропонований для вивчення в підручнику.

Наразі ІКТ впроваджуються не тільки у традиційні уроки, але й частіше за все використовуються для організації позакласної роботи учнів.

На сьогоднішній день найбільш популярними у використанні програмами є: «*Essential Grammar in Use*», «*Britannica*», «*Macmillan English Dictionary*», «*Bridge to English*». [4]

## 1.1. Загальні питання використання ІТ в освіті

В наш час проблема застосування нових ІТ в середній школі є однією із найбільш актуальних. Наразі формування та розвиток інформаційної культури засобами ІМ передбачає посилення уваги на розвиток комунікативно-когнітивних навичок в процесі вивчення цього предмету.

Новітні технології в педагогіці, такі, як використання нових ІТ, Інтернет-ресурсів дозволяють застосовувати особистісно-орієнтований підхід в НП, що сприяє індивідуалізації навчання урахувавши здібності дітей, їх рівень навченості, інтересів і т.п.

Стрімке впровадження інформаційних процесів в різні сфери життя потребує формування новітньої моделі системи освіти на основі сучасних ІТ. Мається на увазі створення умов для розкриття творчого потенціалу людини, розвитку здібностей, виховання потреби самовдосконалення та відповідальності.

Таким чином, впровадження ІКТ допомагає досягнути мети модернізації освіти – покращити якість навчання, збільшити доступність освіти, сприяти розвитку гармонійної особистості, яка може орієнтуватися в інформаційному просторі, залучена до інформаційно-комунікативних можливостей новітніх технологій та володіє інформаційною культурою.

Основними цілями застосування ІКТ на уроках ІМ:

- підвищення вмотивованості вивчення мови;
- розвиток мовної компетенції: вміння розуміти автентичні тексти іноземною мовою, а також вміння передавати інформацію зв'язними аргументованими висловлюваннями;
- збільшення об'єму лінгвістичних знань;
- розширення знань про соціокультурну специфіку країни ІМ, що вивчається;
- розвиток здібностей та готовності до самостійного вивчення ІМ.

Як було сказано раніше, виділяють ряд дидактичних задач, які вирішуються в процесі викладання ІМ за допомогою ІКТ:

- формування та вдосконалення мовних навичок, вмінь читання, письма, аудіювання та мовлення;
- розширення активного та пасивного словників;
- набуття культурологічних знань;
- формування культури спілкування;
- формування елементів глобального мишлення;
- розвиток вмотивованості до пізнавальної діяльності, необхідності використання ІМ з метою живого спілкування;
- формування навичок групової роботи. [4]

Розглянемо основні напрями в яких використовують ІКТ на уроках (за класифікацією Войтко С.А.):

- Використання готових мультимедійних продуктів та КНС.
- Створення власних мультимедійних презентацій.
- Використання ІКТ для позакласної роботи.
- Користування ресурсами мережі Інтернет.

Використання ІКТ в навчанні, в тому числі, вивченні ІМ значною мірою вплинуло на концепцію розробки навчальних матеріалів за цією дисципліною. Інтерактивне навчання на основі навчальних КП дозволяє більш вичерпно синтезувати методичні, дидактичні, педагогічні та психологічні принципи, а також позитивно впливає на процес пізнання, формує його більш цікаво та творчо, дозволяє врахувати для кожного учня його індивідуальний темп роботи. [5]

Сьогодні цілі та задачі, що поставлені перед новітньою освітою змінюються – акцент із засвоєння знань переноситься на формування компетентності. Відбувається переорієнтація на особистісно-орієнтований підхід, що є цілком протилежним до знання-орієнтованої безособистісної педагогіки. Школи забезпечуються сучасними комп'ютерами, електронними ресурсами, доступом до Інтернету. Це сприяє впровадженню нових педагогічних технологій у НП. Саме застосування інноваційних технологій на уроках ІМ є основною ознакою



позитивних результатів творчої діяльності, що несе за собою підвищення вмотивованості учнів. [6]

## 1.2. Створення комп'ютерних засобів для навчання

На даний час існує велике різноманіття сучасних мультимедійних підручників, де можна знайти достатню кількість вправ для тих, хто вивчає ІМ, будь-якого віку та різного рівня знань. Використання КТ на уроках ІМ підвищує мотивацію та пізнавальну активність учнів, розширює їх кругозір. Неоціненну допомогу у вивченні мови надають електронні посібники, бібліотеки, галереї.

Можливості мультимедійних технологій дозволяють учню зануритися у віртуальний світ та побачити те, що може бути написано лише у книжці. Електронні галереї дозволяють проглядати витвори мистецтва великих художників, не зрушуючи з місця, а бібліотеки країн дозволяють відвідати всі місця на земному шарі. Все це знадобиться не лише на уроці, але й в подальшому житті.

Як відомо, комп'ютер має можливість зберігання та використання великої кількості інформації, що використовується для учбової діяльності: різноманітні тексти, вправи, аудіо та мультимедійна інформація. В процесі використання дана інформація може бути змінена, доповнена та переформатована. Застарілий матеріал можна замінити новим, вправи та тексти представити в різноманітних варіантах. Можливості подання навчальної інформації значно розширюються при використанні комп'ютера. Проте, для реалізації всіх цих переваг використання ІКТ необхідно продумати та створити власну комп'ютерну навчальну програму. Будь-яка програма може бути задана за певним сценарієм, що був придуманий вчителем або складений згідно з навчально-методичним комплексом, що використовується в роботі. Всі компоненти комп'ютерної навчальної програми повинні, перш за все, відповідати цілям вивчення ІМ.

Однак аналіз існуючих програмних продуктів продемонстрував, що велика їх кількість є неефективною, оскільки всі види вправ зводяться лише до вибору

правильної відповіді та заповнення пропусків. Також ще одним важливим недоліком таких програм є відсутність можливості змінити зміст програми, оновити та розширити базу даних навчальних текстів доповнити та змінити вправи, з урахуванням рівня знань учнів того чи іншого лексичного матеріалу або відповідно до навчального плану того чи іншого освітнього закладу або конкретного вчителя.

На ринку навчальних КП зростає попит на програмні продукти, в яких можна відзначити такі компоненти: розділ із текстовою інформацією (для надання та редагування тематичного тексту, який вчитель може змінювати та доповнювати на власний погляд), розділ завдань та вправ, що спрямовані на закріплення вивченого матеріалу, граматичних конструкцій та лексичних одиниць, довідковий розділ, що включає в себе як інформацію по роботі із програмою, так і різні словники та граматичні довідники. Також бажано, щоб був передбачений розділ перевірки знань учнів (тестування та автоматична обробка результатів), а також розділ статистичної інформації, в якому би відображалася динаміка роботи учня, а саме прочитані тексти, виконані завдання, рівень сформованості тих чи інших знань, що виражений числовими показниками. Таким чином, вчитель створює програму саме для себе та для своїх учнів, програму за якою йому буде зручно працювати, яка допоможе досягнути поставлених педагогом цілей та здобути високі результати при навчанні АМ. [6]

Наразі існує великий вибір програм, що можна використовувати при вивченні ІМ. Проте, всі програми можна поділити на такі типи:

– Програма для роботи із словником. Такі програми призначені для самостійного створення словника. Користувач може записати певну кількість слів з їх визначеннями та прикладами. В будь-який час він може подивитися слово або перевірити свої знання лексики: наприклад, на екрані демонструється визначення слова та питання про нього. Такі програми дуже прості в користуванні та є ідеальним варіантом не тільки для індивідуального навчання, а також для занять в групі та обміну інформацією.

– Програма на підбір відповідностей. Такі КП універсальні та мають

постійний попит серед користувачів. Завдання – користувач повинен правильно з'єднати лексичні одиниці (від слова до абзацу в декілька рядків) правої та лівої колонок. Вибір відбувається шляхом розгляду двох колонок з використанням стрілочок-індикаторів. Друкуючий пристрій дозволяє розмножувати вправи на папері для індивідуальної та класної роботи.

– Програма на вибір варіантів. Такі прості в користуванні КП призначені для створення вправ на множинний вибір. Програміст може встановити 3-5 варіантів вибору або різну кількість для кожного прикладу та за необхідності підключити індикатори помилок, що реєструють кожну неправильну відповідь. Такі програмні продукти зустрічаються у двох варіантах. Перший, так званий, «Перевіряючий режим» – учень до кінця виконання вправи не знає, які з його відповідей правильні. Другий – «Навчальний режим» – забезпечує негайну відповідь та дію індикатора помилок. В обох випадках за необхідності результати роботи учня (кількість помилок) можуть бути виведені в друкованому вигляді.

– Програма на заповнення пропусків. В програмах такого типу учневі пропонується заповнити в тексті пропуски вручну або вибрати з переліку запропонованих слів. Є також програми, які дозволяють викладачу або учню вводити або редагувати текст. Потім учень обирає текст за назвою та встановлює які слова повинні бути пропущені (наприклад, від кожного п'ятого до кожного п'ятнадцятого). Після цього, переглядаючи текст з пропусками заповнює їх. В таких програмах часто тексти можуть бути виведені на друк, щоб учень міг у письмовому вигляді виконати вправу у зручному для себе темпі, після чого звірити результати своєї роботи з даними комп'ютера.

– Програма на заповнення кросворду. В таких програмах пропонується розв'язати кросворд на комп'ютері, проте інколи також у користувача є можливість створювати та зберігати власні кросворди. Програміст може визначити розміри діаграми (від п'яти до п'ятнадцяти порожніх клітинок у будь-якому напрямі) та надрукувати в ній слова та пробіли.

– Програма на підстановку. У таких програмах поданий текст у якому завчасно відібрані слова або вирази, що одне за одним виділяються більш

яскравим кольором. Учень повинен підставити замість них інші слова та вирази.

– Програма для роботи зі словником до тексту. В програмі складається список всіх слів тексту в алфавітному порядку та вказується частотність вживання кожного слова. Список може бути збережений в електронному вигляді або виведений на друк. Використання таких додатків корисне в підготовці лекичних пояснень до певних текстів.

– Програма на вгадування. В таких програмах потрібно відновити тему тексту через мінімум контексту. Тобто, є певна кількість текстів на різні теми. На екрані показується певна кількість тем та одне слово із тексту. Інші слова опущені. Учень повинен використовуючи мінімум інформації здогадатися яка тема із запропонованих підходить до цього тексту. Користувач може відкривати додаткові слова, поки не відгадає тему.

– Програма на швидкість читання. Найчастіше текст в таких програмах демонструється на екрані частинами на проміжку певного періоду часу (учень сам повинен вибрати). Для кожної програми час для повільного читання та для тренування оперативної пам'яті буде відрізнятися. Мета вправи – тренування швидкого читання із загальним розумінням змісту. Після кожного фрагменту або наприкінці тексту можуть бути запропоновані запитання для перевірки розуміння прочитаного. [2]

### 1.3. Аналіз існуючих рішень

Для порівняльного аналізу відібрані програми двох типів: ті, які працюють локально на комп'ютері та веб-додатки. Такий вибір обумовлений тим, що предметом дослідження є створення веб-додатку на платформі *Unity* і для того, щоб продукт був якісним, необхідно порівняти як працюють вже готові рішення, які особливості, переваги та недоліки мають.

Для порівняльного аналізу обрані такі програми: *Rosetta Stone*, *Tell Me More*, *Bridge to English*, *Duolingo*, *Memrise*, *Babbel*.

Нижче наведено короткий опис для кожного додатку.

*Rosetta Stone* – це, можливо, одна з найкращих програм для вивчення іноземних мов, проте, і найдорожча. За цю ціну користувач отримує мовні уроки вищого класу, які занурять його в англійську мову від початку навчання. У програмі *Rosetta Stone* використовується тільки та мова, яка вивчається, програма спеціально розроблена так, що спочатку темп повільний, а в ході навчання він збільшується, так що у користувача буде досвід занурення в мовне середовище.

Це досить унікальний метод, в порівнянні з іншими. Програма *Rosetta Stone* допоможе вивчати закономірності мови, а не просто запам'ятовувати лексику і правила. У додатку *Rosetta Stone* є вправи, які допоможуть вивчити базову лексику, перш ніж з'явиться необхідність складати фрази і речення. Користувача навчають чомусь одному за раз, і він вчить нові слова, використовуючи знання, які вже має. Додаток допоможе не тільки вчити і ІМ, а й думати на цій мові.

Якщо користувач готовий застосувати на практиці свої вміння та знання, на сервісі *Rosetta Stone* є також онлайн-спільнота і послуги репетиторів. Можна грати в ігри, спілкуватися з іншими користувачами в живому чаті і використовувати свої навички, щоб підвищити рівень знань.

*Rosetta Stone* знаменита своїми передовими та прогресивними мовними технологіями, які використовуються NASA і Держдепартаментом США. Ці ж технології будуть допомагати користувачу читати розповіді і відпрацьовувати вимову.

Наприклад, користувач вчить слова "*man*", "*woman*", "*park*" за допомогою зображень. Наступний крок – речення "*A man and a woman is sitting in the park*". Користувач може ще не знати, що означає слово "*sit*", але, швидше за все, він правильно його вгадає.

За допомогою цього методу з додатком *Rosetta Stone* вдасться вивчити різні слова без використання рідної мови. Це метод занурення, і він дуже дієвий. Ось чому це один з найкращих додатків для вивчення АМ. [7]

*Tell Me More Performance* – чудове ПЗ для вивчення ІМ. *Tell Me More* – простий у використанні та ефективний інтерактивний курс, побудований на основі методик занурення і асиміляції з використанням сучасних технологій і

комунікативного підходу. За даними дослідження *IDC* більше 95% користувачів успішно освоїли принаймні один рівень після 50 годин вивчення. Цей міжнародно визнаний курс використовується більш ніж в 15 000 освітніх установ і компаній по всьому світу.

Оскільки *Tell Me More* скоріше тренажер, ніж підручник, його можна використовувати на різних стадіях освоєння англійської мови. Завдяки комп'ютерним ігровим прийомам, що застосовуються в *Tell Me More*, з програмою не нудно навіть тому користувачеві, який добре володіє мовою. Наприклад, тут реалізований інтерактивний розгалужений діалог – особливістю якого можуть похвалитися далеко не всі навчальні програми.

При навчальній "розмові" з комп'ютером репліка електронного співрозмовника вибирається залежно від репліки користувача. Майже як в житті, розмова може розвиватися по-різному. Наступного разу можна представитися, наприклад, не молодою жінкою, а пенсіонером з двома онуками і подивитися, що заготовлено розробниками на цей випадок.

*Tell Me More Performance English* – це інтерактивний курс вивчення АМ, що можна налаштувати відповідно до цілей навчання і часу, за який необхідно опанувати мову. Програма складає для користувача оптимальний графік занять та покроково, замінюючи викладача і носія мови навчає користувача. Навчальний курс розділений на 10 рівнів складності і буде цікавий як початківцям, так і знавцям мови. Проходження початкового рівня забезпечує впевнене знання основ мови і можливість підтримати бесіду на прості теми. Програма середнього рівня дозволяє навчитися читати художню літературу і вільно спілкуватися англійською мовою у будь-якій побутовій ситуації. Освоєння просунутого рівня дає можливість розуміти на слух теле-, радіопередачі і брати участь в бесіді з носіями мови на будь-яку тему. Повний курс розрахований приблизно на 2000 годин інтенсивних занять. Основою курсу служать інтерактивні багаторівневі діалоги, в яких обігрується поширені ситуації побутового та ділового спілкування. Програма дозволяє вести з комп'ютером вільний діалог, розвиток якого залежить від відповідей учня.

Особливості продукту:

- Розвиток усного та писемного мовлення, навичок читання та аудіювання;
- Технологія розпізнавання мови: можливість введення відповідей за допомогою мікрофона;
- Інтерактивні діалоги з комп'ютером;
- Система візуалізації вимови: відображення мови у вигляді графіка;
- Оцінка вимови і вказівка на неправильно вимовлені слова;
- Анімовані відеоряди, що демонструють артикуляцію звуків;
- Планувальник занять та індивідуальні налаштування. [8]

З *Bridge to English* користувач зможе удосконалювати рівень англійської мови щодня дорогою на роботу або навчання. Навіть довгі автомобільні затори не стануть втраченим часом.

Комплект програм призначений для тих, хто вивчає АМ або для тих, хто прагне вдосконалити навички читання, письма, вимови і розуміння на слух. Розроблено в Великобританії видавництвом Оксфордського університету для професійного використання, він не має собі рівних за ефективністю навчання. Гарантований результат досягається застосуванням численних методичних прийомів і жорсткої системи контролю за успішністю.

В програмі закладена захоплююча і динамічна вправа Тир, в якій необхідно написати потрібне слово за певний відрізок часу і "вистрілити". Так користувач навчається писати без помилок.

У вправах для розвитку мовних навичок користувач має можливість побачити і почути, як англійці спілкуються один з одним. Розробники передбачили також використання традиційного англійського гумору. В програмі використані елементи штучного інтелекту. Програма здатна сприйняти безліч, іноді десятки тисяч варіантів реплік користувача. За допомогою професійних дикторів можна тренувати вимову. В програмі передбачена можливість використовувати різні режими запису та відтворення звуку.

Найскладніше у вивченні мови – навчитися сприймати мову на слух. Цю проблему вирішать прості і складні диктанти, що добре себе зарекомендували.

В програмі вперше використано оригінальне представлення граматичних конструкцій у вигляді змінних штампів. Для закріплення матеріалу слугує мозаїка.

Курс містить звукові доріжки, які можна прослухати на будь-якому програвачі музичних компакт-дисків, плеєрі або на комп'ютері. Звуковий матеріал включає граматичні коментарі російською мовою, а також переклади всіх прикладів використання досліджуваної граматичної конструкції. Завдяки цьому користувач може продовжувати вчити англійську мову без комп'ютера у будь-якому зручному місці.

Завдяки унікальним вправам, англо-російський і англо-англійський тренажери допоможуть у вивченні АМ, поєднуючи корисне з приємним – навчання з розвагою. За вибором користувача, в ігрових вправах можна відгадувати як англійські слова за їх російськими значеннями, так і російські слова за їхнім англійським значенням.

Вправа «Тир» розвиває навички правопису і здатність швидко підбирати потрібні слова у будь-якій ситуації. Користувач може поступово збільшувати словниковий запас, вибираючи набори з 500/1000/1500/2000/2500/3000 найбільш уживаних слів мови. Крім того, користувач може скласти самостійно блокноти для вивчення саме тих слів, які йому найбільш необхідні.

Унікальний алгоритм генератора кросвордів побудує для користувача необмежену кількість головоломок, що ніколи не повторюються. Відгадуючи кросворди, користувач не тільки отримує задоволення, але й істотно збільшує свій словниковий запас, дізнаючись нові слова, а також нові значення вже відомих вам слів.

Результати навчання фіксуються і наочно відображаються за допомогою кольорових діаграм.

Дані комп'ютерні програми працюють на усіх версіях ОС *Windows* та є платними. [9]



Розглянемо додатки, які крім комп'ютерної версії можуть працювати в режимі онлайн.

Розуміючи, що часто мовне ПЗ коштує досить дорого, розробники *Duolingo* розробили безкоштовну програму, яка містить матеріал приблизно одного університетського семестру англійської, який викладається інтерактивним методом.

Переваги програми: *Duolingo* дозволяє опанувати базовий рівень англійської безкоштовно. Отримані результати, оцінюються автоматично, так щоб користувач мав чітке уявлення про досягнуті успіхи. Програма *Duolingo* гейміфіційована: вона нагороджує користувача за успішно пройдені уроки і досягнення певного рівня. Програма *Duolingo* відмінно підійде не тільки для початківців, а й для просунутих користувачів – з нею користувач регулярно повертається назад і повторює пройдений матеріал, тому забути його не вийде. З *Duolingo* можна вивчати англійську будь-якою з безлічі мов (французькою, португальською, російською, італійською, голландською, іспанською), причому команда сервісу обіцяє збільшити в майбутньому список підтримуваних мов.

Завдання *Duolingo* полягає в тому, щоб допомогти вивчити англійську якомога швидше. Це означає, що якщо користувач ще жодного разу в житті не намагався вивчити англійську, то, займаючись з *Duolingo* по 20 хвилин в день, вже досить скоро зможе використовувати в мові прості конструкції англійською мовою, читати і розпізнавати прості фрази англійською. Це дуже ефективний метод. У кожному уроці *Duolingo* знайомить користувача з сімома новими словами на певну тему. У кожному уроці також є вправи – наприклад, де треба зіставити нові слова із зображенням, перевести фразу з англійської на рідну мову або повторити сказану англійською фразу.

Звичайно, не оминати тут увагою і проблему слів, які забуваються. *Duolingo* використовує грамотний підхід: через деякий час (скажімо, тиждень) сервіс пропонує попрактикуватися зі словами, які давалися користувачеві найскладніше. Відповідно, чим більше таких вправ він виконає, тим простіше буде запам'ятати відповідну лексику.

Займаючись з цим невеликим додатком всього по 10-20 хвилин в день можна пройти всі уроки *Duolingo* приблизно за 3 місяці. Додаток можна завантажити на пристрої з ОС *iOS* та *Android*.

*Duolingo* вважається одним із кращих додатків сучасності для вивчення англійської мови, який отримав відмінні відгуки користувачів. [10]

Основне завдання програми *Babbel* – навчити користувача спілкуватися живою розмовною англійською. Додаток розроблено так, щоб їм просто було користуватися кожен день, оскільки курс розбитий на 15-хвилинні уроки. Користувач вчиться і практикується на реалістичних діалогах. У додатку *Babbel* навіть можна записувати і розпізнавати свої висловлювання, щоб виявити помилки вимови. Користувач також може бути впевненим, що запам'ятав увесь матеріал, тому що додаток *Babbel* періодично нагадує повторювати важливі уроки, щоб вони не забувалися.

Однією із кращих особливостей додатку *Babbel* є те, що програма допомагає користувачу знайти уроки, які йому цікаві. Користувач може практикувати англійську за допомогою будь-яких тем, які для нього важливі, від бізнесу до подорожей. Завдяки цьому вчити англійську стає набагато цікавіше, адже таким чином виробляється хороша мотивація. Додаток має онлайн-версію, а також працює на таких ОС як *iOS* та *Android*. Для того, щоб користуватися даним програмним додатком необхідно придбати ліцензію. [10]

*FluentU* перетворює відео з реального світу, такі як музичні кліпи, рекламні ролики, випуски новин та надихаючі виступи, в уроки англійської мови. На відміну від традиційних програм, *FluentU* використовує природний підхід до вивчення англійської мови, який допоможе користувачу з легкістю «влитися» в англійську мову і англійську культуру з часом.

У бібліотеці *FluentU* можна знайти відео на різну тематику: популярні ток-шоу, музичні кліпи і смішну рекламу. З *FluentU* дуже просто дивитися англійське відео. Тут є інтерактивні субтитри. Це означає, що користувач може натиснути на слово і побачити до нього визначення, зображення і корисні приклади. Вивчати лексику з відео можна за допомогою тестів.

Дуже добре, що *FluentU* запам'ятовує лексику, яку вчить користувач, і дає йому рекомендації та приклади відео зі словами, які він вже знає. У кожного користувача будуть свої власні індивідуальні уроки, навіть якщо вони будуть дивитися один і те саме відео. Можна почати вчитися з *FluentU* через веб-сайт за допомогою свого комп'ютера або планшета, а можна завантажити додаток з магазину *iTunes* або *Google Play*. [10]

*Busuu* трохи відрізняється від додатків, які були описані вище. Справа в тому, що велика частина додатків для вивчення англійської мови призначені для особистого використання. Більшу частину уроків у них користувач буде проходити самотійно. Але з *Busuu* є можливість поспілкуватися з носієм АМ і поліпшити навички володіння усною мовою. У кожному уроці *Busuu* користувача чекатимуть тематична лексика, навчальний діалог, письмова вправа, усна вправа, огляд уроку і, звичайно ж, та частина, де користувач зможе поспілкуватися з носієм АМ. Таким чином, можна поліпшити навички читання, письма, спілкування і сприйняття мови на слух. Це дуже збалансований спосіб вивчення англійської мови. Продукт платний, проте демо-версія продукту безкоштовна. Додаток доступний для користування онлайн та для мобільних пристроїв на ОС *iOS* та *Android*. [10]

#### 1.4. Висновки до розділу

При написанні даного розділу виявлено, що використання інноваційних інформаційних технологій на уроках ІМ позитивно впливає на рівень вмотивованості учнів.

Наразі існує велика кількість мультимедійних підручників, де можна знайти достатню кількість вправ для тих, хто вивчає ІМ, будь-якого віку та різного рівня знань. Використання КТ на уроках ІМ підвищує мотивацію та пізнавальну активність учнів, розширює їх кругозір. Неоціненну допомогу у вивченні мови надають електронні посібники, бібліотеки, галереї.

Можливості мультимедійних технологій дозволяють учню зануритися у віртуальний світ та побачити те, що може бути написано лише у книжці. Електронні галереї дозволяють побачити витвори мистецтва великих художників, не зрушуючи з місця, а бібліотеки країн дозволяють відвідати всі місця на земному шарі. Все це знадобиться не лише на уроці, але й в подальшому житті.

Порівнюючи готові програмні рішення було виявлено, що на сучасному ринку лінгвістичного програмного забезпечення, програмні продукти розраховані на будь-який рівень знання мови та передбачені в них різних типів вправи для якісного тренування того чи іншого аспекту мови. Також можна помітити таку тенденцію: коли навчальні КП тільки починали створюватися, то здебільшого вони були розраховані для ПК та ОС *Windows*, проте всього за 10-15 років все змінилося та наразі акцент ставиться на веб-версії програмних додатків, щоб використання програми мінімально залежало від технічних характеристик комп'ютера, та ОС *iOS* і *Android*, щоб користувач у будь-який зручний час міг вивчати ІМ маючи під рукою лише свій телефон або планшет.

Постановка задачі – в ході опису вже існуючих програмних рішень для написання власного програмного модулю було поставлено такі завдання:

- проектування діаграм ідентифікації точок зору;
- розробка схем алгоритмів;
- *UML*-опис;
- аналіз методів розробки програмного модуля;
- розробка програмного модулю;
- тестування готового програмного продукту;
- написання інструкцій користувача.

## РОЗДІЛ 2

### ОБГРУНТУВАННЯ ПІДХОДУ ДО РОЗРОБКИ ПРОГРАМНОГО МОДУЛЮ

При аналізі задачі та предметної області було вирішено, що для зручного користування, додаток повинен працювати в режимі онлайн та підтримувати кроссбраузерність, адже так користувачі із різними операційними системами зможуть запустити програму, просто оновивши браузер до останньої версії.

Вибір платформи *Unity* для написання браузерної гри вмотивований тим, що на відміну від мови програмування *JavaScript*, фреймворків *Angular* та *React* або різних графічних бібліотек, таких як, наприклад, *Three.js*, що призначені для написання веб-додатків, перший варіант дозволяє легко управляти графікою додатку або налаштовувати анімацію та у будь-який момент, за необхідності, створити версію гри для локального використання на комп'ютері, телефоні (ОС *Android* або *iOS*) або іншій платформі. Так, наприклад, технічні можливості *Windows Forms Application* не дозволяють створити додаток, що зможе працювати онлайн.

Нижче детально описані технології, які використовуються для написання цільового додатку, наведені графічні матеріали, що демонструють роботу програми.

#### 2.1. Платформа *Unity*

*Unity* – це крос-платформна ігрова платформа, розроблена *Unity Technologies*, вперше випущена у червні 2005 року на всесвітній конференції розробників *Apple Inc.* як ексклюзивна ігрова платформа для *Mac OS X*. З тих пір платформа поступово розширюється для підтримки різноманітних настільних, мобільних, консольних та віртуальних додатків.

Кафедра КСУ				НАУ 21 20 53 000 ПЗ			
Виконав	Іванова М. С.			Обґрунтування підходу до розробки програмного модулю	Літера	Аркуш	Аркушів
Керівник	Апенько Н. В.					28	61
Консульт.					СП-435 123		
Норм. контр.	Тупота Є. В.						
Зав. Каф.	Литвиненко О.Є.						

Платформа особливо популярна для розробки мобільних ігор для *iOS* та *Android* і використовується для таких ігор, як *Pokemon Go*, *Monument Valley*, *Call of Duty, Mobile*, *Beat Sabre* та *Cuphead*. Вважається простою у використанні для розробників-початківців і популярна для розробки ігор *Indie*. [11]

Платформу можна використовувати для створення тривимірних (*3D*) та двовимірних (*2D*) ігор, а також інтерактивних симуляцій. Платформа була прийнята для розробки в інших сферах за межами відеоігор, таких як кіно, автомобільна промисловість, архітектура, машинобудування та будівництво.

З моменту запуску було випущено кілька основних версій *Unity*. Остання стабільна версія, 2021.1.0, була випущена 23 березня 2021 року.

*Unity* надає користувачам можливість створювати ігри як у *2D*, так і в *3D*, а також пропонує основний *API* сценаріїв на *C#* як для редактора *Unity* у формі плагінів, так і для самих ігор. До того, як *C#* була основною мовою програмування, що використовувалася для платформи, він раніше підтримував *Boo*, який був видалений випуском *Unity5*, та версію *JavaScript* під назвою *UnityScript*, яка була припинена в серпні 2017 року після випуску *Unity 2017.1*, на користь *C#*.

У *2D*-іграх *Unity* дозволяє імпортувати спрайти та вдосконалений *2D*-світовий візуалізатор. Для *3D*-ігор *Unity* дозволяє задавати стиснення текстур, міпмапи (*MipMaps*) та налаштування роздільної здатності для кожної платформи, яку підтримує ігровий рушій і забезпечує підтримку відображення ударних зображень, паралакса, оклюзії навколишнього середовища екрану, динамічні тіні за допомогою карт тіней, ефектів візуалізації текстури та повноекранної обробки.

Редактор *Unity* підтримується на *Windows*, *macOS* та платформі *Linux*, тоді як сама платформа в даний час підтримує створення ігор для більш ніж 19 різних платформ, включаючи мобільні, настільні, консольні та віртуальну реальність. Офіційно підтримуваними платформами для *Unity 2020 LTS* є:

- мобільні платформи *iOS*, *Android (Android TV)*, *tvOS*;
- настільні платформи *Windows* (Універсальна платформа *Windows*), *Mac*, *Linux*;

- веб-платформа *WebGL*;
- консольні платформи *PlayStation (PS4, PS5), Xbox (Xbox One, Xbox Series X/S), Nintendo Switch, Stadia*;

Платформи віртуальної / розширеної реальності *Oculus, PlayStation VR, Google ARCore, Apple ARKit, Windows Mixed Reality (HoloLens), Magic Leap, та через Unity XR SDK Steam VR, Google Cardboard*. [13]

Раніше підтримуваними платформами були *Wii, Wii U, PlayStation 3, Xbox 360, Tizen, PlayStation Vita, 3DS, BlackBerry 10, Windows Phone 8, Samsung Smart TV, Gear VR, Daydream, Vuforia, та Facebook Gameroom*.

Станом на 2018 рік платформа *Unity* використовувалася для створення приблизно половини мобільних ігор на ринку та 60 відсотків вмісту доповненої реальності та віртуальної реальності, включаючи приблизно 90 відсотків на нових платформах доповненої реальності, таких як *Microsoft HoloLens*, і 90 відсотків вмісту *Samsung Gear VR*. Технологія *Unity* є основою для більшості програмного забезпечення віртуальної реальності та доповненої реальності, і *Fortune* сказала, що *Unity* "домінує у бізнесі віртуальної реальності".

Агенти машинного навчання *Unity* – це програмне забезпечення з відкритим кодом, завдяки якому платформа *Unity* підключається до програм машинного навчання, включаючи *Google TensorFlow*. Програмне забезпечення використовується для розробки роботів та самокерованих автомобілів.

Раніше рушій *Unity* підтримував інші платформи, включаючи власний веб-плеєр *Unity*, плагін веб-браузера. Однак його використання припинено на користь *WebGL*. Починаючи з версії 5, *Unity* пропонує свій пакет *WebGL*, скомпільований до *JavaScript*, використовуючи двоступеневий мовний перекладач (*C#* на *C++* і, нарешті, *JavaScript*).

## 2.2. Особливості обудови онлайн-додатку за допомогою *WebGL*

Параметр компіляції проєкту *WebGL* дозволяє *Unity* публікувати програми, які використовують *HTML5/JavaScript*, *WebAssembly*, *API WebGL* та інші веб-стандарти для запуску проєктів *Unity* у веб-браузері.

Додаток *Unity WebGL* підтримується більшістю основних браузерів, однак існують відмінності в рівні підтримки, що пропонується різними браузерами. Мобільні пристрої не підтримують *Unity WebGL*.

Не всі функції *Unity* доступні у *WebGL*, здебільшого через обмеження платформи. Потоки не підтримуються через відсутність підтримки потоків у *JavaScript*. Це стосується як внутрішнього використання потоків *Unity* для пришвидшення продуктивності, так і використання потоків у коді сценарію та керованих *dll*. Проєкти *WebGL* неможливо відладити у *Visual Studio*. Браузери не дозволяють прямий доступ до *IP*-сокетів для роботи в мережі, через проблеми безпеки. Графічний *API WebGL* еквівалентний *OpenGL ES 2.0* і *3.0*, що має деякі обмеження. За замовчуванням проєкти *Unity WebGL* підтримують як *API WebGL 1.0*, так і *WebGL 2.0*. Це можна змінити в налаштуваннях плеєра *WebGL* на панелі «Інші налаштування», вимкнувши властивість *Automatic Graphics API*. Коли в браузерах підтримується *WebGL 2.0*, додаток може бути кращої якості у стандартному шейдері, є підтримка інстансування графічного процесора, відсутність обмежень щодо індексації та циклів у шейдерному коді та кращу продуктивність.[12]

У проєктах *WebGL* використовується власний сервер для аудіо на основі *API* веб-аудіо. Платформа підтримує лише базові звукові функції. *WebGL* не дозволяє динамічно генерувати код за допомогою *System.Reflection.Emit*. *Unity WebGL* підтримують всі основні браузери для настільних комп'ютерів. Однак рівень підтримки та очікувана продуктивність можуть бути різними.

Слід звернути увагу, що додатки *Unity WebGL* наразі не підтримується на мобільних пристроях, тому що багато сучасних пристроїв недостатньо потужні та не мають необхідного об'єму пам'яті. З цієї причини *Unity WebGL* відображає



попереджувальне повідомлення при спробі завантажити вміст у мобільні браузері (яке за потреби можна відключити).

Збірки *.NET* можуть містити вбудовані ресурси. Вбудований ресурс – це колекція двійкових даних, яка є частиною проєкту *.NET*. Користувач може отримати доступ до цих двійкових даних у кодї за допомогою файлоподібного *API*. За замовчуванням *WebGL* проєкти не включають вбудовані ресурси. Це допомагає зменшити розмір двійкового файлу, оскільки вбудовані ресурси можуть бути досить великими. Однак деякі користувацькі коди та *API*-інтерфейси бібліотеки класів *.NET* залежать від наявності вбудованих ресурсів для належної роботи.

Коли користувач створює *WebGL* проєкт, *Unity* вбудовує плеєр у *HTML*-сторінку, щоб браузер міг його відкрити. Шаблон *WebGL* – це параметр конфігурації, який дозволяє контролювати вигляд *HTML*-сторінки, що дає можливість тестувати, демонструвати та попередньо переглядати свою програму.

З міркувань безпеки браузері дозволять блокувати курсор або переходити в повноекранний режим лише у відповідь на ініційовану користувачем подію (наприклад, клацання миші або натискання клавіші). На жаль, *Unity* не має окремих подій та візуалізації, тому він відкладає обробку подій до моменту, коли браузер більше не розпізнає повноекранний запит або запит на блокування курсору, виданий сценарієм *Unity*, як пряму відповідь на подію, яка його спричинила. Як результат, *Unity* ініціює запит на наступну ініційовану користувачем подію, а не на подію, яка спричинила блокування курсору або запит повноекранного режиму.[14]

Щоб зробити цю роботу з прийнятними результатами, потрібно запускати блокування курсору або повноекранний режим, коли кнопка миші натиснута, замість події миші, коли кнопку відпустили. Це гарантує, що коли запит переноситься на наступну ініційовану користувачем подію (коли користувач відпускає мишу або клавішу) – цільова команда виконується. Якщо у налаштуваннях плеєра для платформи *WebGL* увімкнено функцію «Запуск у фоновому режимі» або якщо користувач ввімкнув *Application.runInBackground*,

додаток продовжує працювати, навіть коли на вікні браузера або вкладці немає фокусу.

Якщо є необхідність використовувати компонент *Unity UI.Button*, то можна досягти бажаної поведінки, створивши підклас *Button*, який замінює метод *OnPointerDown*. Слід звернути увагу, що браузери можуть показати користувачу повідомлення з питанням про дозвіл перед входом у повноекранний режим або блокуванням курсору. Геймпади та джойстики підтримуються в *WebGL* (за допомогою класу *Input*) у браузерах, які підтримують *API HTML5 Gamepad*.

За замовчуванням, *Unity WebGL* обробляє всі події з клавіатури, незалежно від того, має вікно *WebGL* фокус чи ні. Це робиться для того, щоб користувач міг негайно розпочати гру з клавіатури, без необхідності натискати на вікно, щоб сфокусуватися спочатку на ньому. Однак це може спричинити проблеми, коли на сторінці є інші елементи *HTML*, які повинні отримувати введення з клавіатури, наприклад текстові поля, оскільки *Unity* буде обробляти введені події, перш ніж їх зможе отримати решта сторінки. Якщо потрібно, щоб інші елементи *HTML* отримували введення з клавіатури, можна змінити цю поведінку за допомогою властивості *WebGLInput.captureAllKeyboardInput*. [15]

*WebGL graphics API* використовує графічний процесор для апаратно-прискореної візуалізації. На центральному процесорі *Emscripten* перекладає код у *WebAssembly*, тому продуктивність залежить від веб-браузера.

Однак деякі браузери можуть регулювати процеси, що виконуються на фонових вкладках. В більшості браузерів контент на вкладці оновлюється лише раз на секунду. Це призводить до того, що *Time.time* працює повільніше, ніж зазвичай, із налаштуваннями за замовчуванням, оскільки значення за замовчуванням *Time.maximumDeltaTime* менше, ніж одна секунда.

Можливо, користувачеві захочеться ввімкнути додаток *WebGL* із меншою частотою кадрів, щоб зменшити навантаження на центральний процесор. Як і на інших платформах, для цього можна використовувати *API Application.targetFrameRate*. [16]

Якщо користувач не хоче знижувати продуктивність, встановити для цього *API* значення за замовчуванням мінус 1 замість високого значення. Це дозволяє браузеру налаштувати частоту кадрів для найбільш плавної анімації в циклі візуалізації браузера.

Додаток *WebGL* працює в браузері. Браузер виділяє пам'ять у своєму просторі пам'яті, що потрібна програмі для запуску. Обсяг доступної пам'яті залежить від того який пристрій, операційна система, браузер та процесор, на якому це все працює (32 або 64-бітному) використовуються.

Незалежно від того, використовує браузер окремі процеси для кожної вкладки, чи додатку потрібно спільно використовувати простір пам'яті з усіма іншими відкритими вкладками. *Unity* використовує купу для зберігання всіх об'єктів середовища роботи платформи. Купа *Unity* – це суміжний блок виділеної пам'яті. Платформа підтримує автоматичне змінення розміру купи відповідно до потреб програми. Розмір купи розширюється під час запуску програми та може досягати 2 Гб. *Unity* створює цю купу пам'яті як об'єкт пам'яті. Властивість буфера об'єкта *Memory* – це змінний *ArrayBuffer*, який містить необроблені байти пам'яті, доступ до яких здійснюється за допомогою коду *WebAssembly*. [11]

Автоматичне змінення розміру купи може спричинити збій програми, якщо браузеру не вдається виділити суміжний блок пам'яті в адресному просторі. З цієї причини важливо зробити розмір купи *Unity* якомога меншим. Користувачу потрібно бути уважним при плануванні використання пам'яті програми.

Коли користувач створює проєкт *Unity WebGL*, платформа генерує файл даних. Тут містяться всі сцени та асети, які потрібно запусити програмі. Оскільки *Unity WebGL* не має доступу до реальної файлової системи, він створює віртуальну файлову систему в пам'яті, і браузер розпаковує файл *.data*. Фреймворк *Emscripten (JavaScript)* розподіляє цю файлову систему у просторі пам'яті браузера. Поки програма працює, пам'ять браузера зберігає нестиснуті дані. Щоб як час завантаження, так і використання пам'яті були низькими, потрібно намагатися, щоб ці нестиснені дані були якомога меншими.

Щоб зменшити використання пам'яті, можна запакувати дані в *AssetBundles*. *AssetBundles* дозволяють повністю контролювати завантаження. Це означає, що користувач може контролювати, коли програма завантажує дані, а коли вивантажує. Вивантаження невикористаних даних звільняє пам'ять. *AssetBundles* завантажуються безпосередньо в купу *Unity*, тому вони не призводять до додаткового розподілу з боку браузера.[12]

Увімкнувши кешування даних, можна автоматично кешувати дані об'єкта додатку на комп'ютері користувача. Це означає, що не потрібно повторно завантажувати ці дані під час наступних запусків. Завантажувач *Unity WebGL* реалізує кешування даних за допомогою *API IndexedDB*. Ця опція дозволяє кешувати файли, які занадто великі, щоб браузер міг самостійно виконувати кешування. Замість того, щоб зберігати вміст у кеші браузера, *Unity* зберігає дані в *IndexedDB*. Браузери часто обмежують кількість, яку можна зберегти в кеш-пам'яті, і максимальний розмір файлу, який можна кешувати. Цього часто недостатньо, щоб програма працювала безперебійно. Сервер може видавати заголовок *HTTP* з великим розподілом для програми. Дане повідомлення можуть отримувати підтримувані браузери (наразі лише *Mozilla Firefox*) про потреби в пам'яті. Ця інформація дозволяє підтримуваному браузеру породити новий процес, використовуючи нефрагментований простір пам'яті. Браузер також може перевіряти, що великий розподіл успішний. Такий підхід може вирішити проблеми, коли у браузера закінчується пам'ять при спробі розподілити купу *Unity*. Це особливо важливо для 32-розрядних браузерів.[14]

Збір сміття – це процес пошуку та звільнення невикористаної пам'яті. Після того, як збирач сміття збирає невикористану пам'ять, він перерозподіляє її всередині купи *Unity*. Збір сміття *WebGL* запускається, коли стек порожній. Стэк є частиною купи *Unity*, але не купи загалом. Це відрізняється від процесу збору сміття на інших платформах, де призупиняються всі запущені потоки, щоб перевірити стек, потім перевіряються стеки потоків та реєструються завантажені посилання на об'єкти. Це неможливо в *JavaScript*. З цієї причини збирач сміття буде працювати в *WebGL* лише в ситуаціях, коли стек порожній. В даний час це

відбувається один раз після кожного кадру. Проте можна налаштувати процес збору сміття за допомогою *Unity Profiler*.

*Visual Studio* не підтримує відладку проєктів *Unity WebGL*. Щоб точно з'ясувати, що відбувається з програмою, нижче приведено декілька рекомендацій, як отримати інформацію.

*Unity WebGL* не має доступу до файлової системи, тому в готовому проєкті немає файлу журналу, як інші платформи. Однак він записує всю інформацію про реєстрацію (наприклад, *Debug.Log*, *Console.WriteLine* або *Unity* для внутрішнього журналювання) у консоль *JavaScript* браузера. Як варіант можна створити збірку проєкту в *Unity* (відкрити вікно Налаштування збірки та натиснути прапорець Розробка збірки). Збірки розробок дозволяють підключити профайлер, а *Unity* не зменшує їх, тому випущений код *JavaScript* все ще містить зручні для читання імена функцій. Браузер може використовувати їх для відображення слідів стека, коли користувач стикається з помилкою браузера, під час використання *Debug.LogError* або коли видаляє виключення і підтримка виключень вимкнена. На відміну від керованих стеків, які можуть виникати, коли їх повна підтримка вимкнена, ці результати трасування стека містять не лише керований код, але й внутрішній код *UnityEngine*.

*WebGL* має різні рівні підтримки виключень. За замовчуванням *Unity WebGL* підтримує лише явно виявлені виключення. Можна ввімкнути повну підтримку виключень, яка виконує додаткові перевірки в коді, згенерованому *IL2CPP*, для отримання доступу до нульових посилань та елементів масиву поза межами керованого коду. Ці додаткові перевірки суттєво впливають на продуктивність та збільшують розмір коду та час завантаження, тому використовувати його слід лише для відладки. Повна підтримка виключень також видає імена функцій, щоб генерувати сліди в стеку для керованого коду. З цієї причини в консолі з'являються повідомлення стека для виключень, які не були відновлені та для інструкцій *Debug.Log*. Щоб отримати рядок трасування стека, необхідно використовувати *System.Environment.StackTrace*. [15]

З міркувань безпеки код *JavaScript* не має прямого доступу до *IP*-сокетів для реалізації мережевого підключення. Як результат, мережа класів *.NET* (тобто все в просторі імен *System.Net*, особливо *System.Net.Sockets*) не працюють у *WebGL*. Те саме стосується старих класів *UnityEngine.Network*, які недоступні під час створення *WebGL*. Якщо є необхідність використовувати мережу в *WebGL*, на даний момент є можливість використовувати класи *WWW* або *UnityWebRequest* в *Unity* або нові функції *Unity Networking*, які підтримують *WebGL*, або реалізувати власні мережі за допомогою *WebSockets* або *WebRTC* в *JavaScript*. Класи *WWW* та *UnityWebRequest* підтримуються в *WebGL*. Вони реалізовані за допомогою класу *XMLHttpRequest* у *JavaScript*, використовуючи браузер для обробки запитів *WWW*. Це накладає певні обмеження безпеки на доступ до міждоменних ресурсів. В основному будь-який запит *WWW* до сервера, який відрізняється від сервера, що розміщує проєкт *WebGL*, повинен бути авторизований сервером, до якого користувач намагається отримати доступ. Щоб отримати доступ до міждоменних ресурсів *WWW* у *WebGL*, сервер, до якого необхідно отримати доступ, повинен авторизувати це за допомогою *CORS*. Якщо користувач намагається отримати доступ до вмісту за допомогою *WWW* або *UnityWebRequest*, а на віддаленому сервері не налаштовано або налаштовано *CORS* неправильно, в консолі браузера з'явиться така помилка: «*Cross-Origin Request Blocked: The Same Origin Policy disallows reading the remote resource at http://myserver.com/. This can be fixed by moving the resource to the same domain or enabling CORS*». В основному, сервер повинен додати деякі заголовки контролю доступу до відправлених відповідей *http*, які повідомлять браузерам, що дозволено веб-сторінкам отримувати доступ до даних на сервері. Ось приклад налаштування заголовка, який дозволить *Unity WebGL* отримувати доступ до ресурсів на веб-сервері з будь-якого джерела, із загальними заголовками запитів та за допомогою методів *http GET, POST* або *OPTIONS*:

*"Access-Control-Allow-Credentials": "true",*

*"Access-Control-Allow-Headers": "Accept, X-Access-Token, X-Application-Name, X-Request-Sent-Time",*

*"Access-Control-Allow-Methods": "GET, POST, OPTIONS",*

*"Access-Control-Allow-Origin": "\*"*

Слід звернути увагу на те, що *WWW.responseHeaders* обмежується підмножиною фактичних заголовків відповідей, відповідно до 7.1.1 специфікації *CORS*. Також необхідно зауважити, що *XMLHttpRequest* не дозволяє потокове передавання даних, тому клас *WWW* на *WebGL* оброблятиме будь-які дані лише після завершення завантаження (тому *AssetBundles* не можна розпаковувати та завантажувати під час завантаження, як на інших платформах). Не потрібно використовувати код, який блокує завантаження *WWW* або *WebRequest*, наприклад: *while(!www.isDone) {}*. [16]

Блокування завантажень *WWW* або *WebRequest* не працює на *Unity WebGL*. Оскільки *WebGL* є однопотокним і оскільки клас *XMLHttpRequest* у *JavaScript* є асинхронним, завантаження ніколи не закінчується, якщо не повернути контроль у браузер. Замість цього краще скористатися *Coroutine* та оператором *yield*, щоб дочекатися завершення завантаження. *Unity Networking* підтримує *WebGL*, дозволяючи зв'язок через протокол *WebSockets*. Як писалося вище, прямий доступ до *IP*-сокетів неможливий у *WebGL*. Однак, якщо користувачу потрібні мережеві можливості за межами класу *WWW*, можна використовувати *Web Sockets* або *WebRTC*, обидва мережеві протоколи, що підтримуються браузерами. *Web Sockets* має ширшу підтримку, але *WebRTC* дозволяє *p2p*-з'єднання між браузерами та ненадійними з'єднаннями. Жоден з цих протоколів поки не доступний через вбудовані *API* в *Unity*, але для реалізації цього можна використовувати плагін *JavaScript*. Зразок плагіна, що реалізує мережу *WebSocket* у *Unity WebGL*, можна знайти на *AssetStore*. [12]

### 2.3. *IL2CPP* та *Emscripten*

Технологія *IL2CPP* складається з двох частин:

- компілятор *Ahead-of-time (AOT)*;
- виконуюча бібліотека для підтримки віртуальної машини.

*AOT*-компілятор переводить проміжну мову (*IL*) з *.NET*-компіляторів в вихідний код *C++*. Виконувана бібліотека надає сервіси і абстракції (такі як збирач сміття), міжплатформений доступ до потоків вводу і файлів, а також способи реалізації внутрішніх викликів (некерований код, безпосередньо змінює керовані структури даних). *AOT*-компілятор *IL2CPP* називається *il2cpp.exe*. У *Windows* його можна знайти в директорії *Editor \ Data \ il2cpp*. Утиліта *il2cpp.exe* повністю написана на мові *C#* і скомпільована за допомогою *.NET* і компіляторів *Mono*. Ця утиліта приймає керовані збірки, скомпільовані *Mono*-компілятором, що поставляються з *Unity*, і генерує код *C++*, який передається в компілятор *C++* для конкретної платформи.[11]

Інструментарій *IL2CPP* можна схематично уявити так (рис. 2.1):

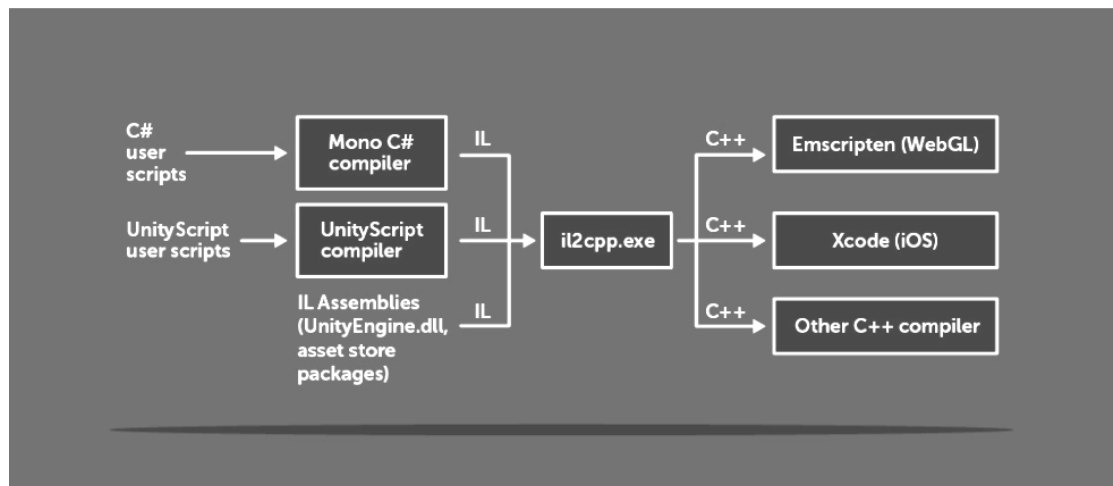


Рис. 2.1. Інструментарій *IL2CPP*

Друга частина технології *IL2CPP* – це виконувана бібліотека для підтримки віртуальної машини. Написана ця бібліотека майже повністю на *C++* і має назву *libil2cpp*. Вона поставляється у вигляді статичної бібліотеки, пов'язаної з



виконуваним файлом плеєра, а її простота і портативність є однією з ключових переваг технології *IL2CPP*.

Користувач можете отримати більш чітке уявлення про організацію коду *libil2cpp*, дивлячись на файли заголовків, що поставляються з *Unity* (їх можна знайти в директорії *Editor \ Data \ PlaybackEngines \ webgl-support \ BuildTools \ Libraries \ libil2cpp \ include* в *Windows*). Наприклад, інтерфейс між кодом *C++*, що генерується *il2cpp.exe*, і середовищем *libil2cpp* знаходиться в файлі заголовка *codegen / il2cpp-codegen.h*. [15]

Один з ключових елементів середовища – збирач сміття. У комплект *Unity5* входить *libgc*, збирач сміття *Boehm-Demers-Weiser*. Однак *libil2cpp* підтримує і інші збирачі.

Хотілося б звернути увагу на одну складність, яку розробники вирішили обійти стороною і таким чином уникли багатьох проблем. Вони не намагалися переписати стандартну бібліотеку *C#* під *IL2CPP*. При створенні проекту *Unity* з використанням *IL2CPP* весь код стандартних бібліотек *C#* (*mcorlib.dll*, *System.dll* і т.д.) є точною копією коду *Mono*-скриптів. Розробники використали код стандартних бібліотек *C#*, тому що він вже встиг добре зарекомендувати себе в багатьох проектах. Таким чином, якщо десь виникає помилка, можна бути впевненим, що проблема в *AOT*-компіляторі або виконуваний бібліотеці.

*Emscripten* — компілятор типу код-у-код або транскompілятор. На вході він приймає *LLVM*-байткод, отриманий компіляцією коду мовою *C* або *C++*. На виході він повертає файл з кодом мовою *JavaScript*, що може бути запущений у веб-плеєрі.

Засновником проекту є Алон Закай. Поширюється на умовах ліцензії *MIT*.

Основною метою розвитку проекту *Emscripten* є створення інструменту, який дозволив би виконувати у браузері код незалежно від мови програмування, якою цей код написаний. *Emscripten* дозволяє здійснити компіляцію біткода *LLVM* в скрипт мовою *JavaScript*, який може бути виконаний всередині веб-браузера, використовуючи тільки штатний *JavaScript*-рушій, без необхідності задіяння додаткових плагінів. Біткод *LLVM* може бути згенерований з *C/C++* за

допомогою компіляторів *lvm-gcc* і *clang*, а також з коду на будь-якій іншій мові програмування, для якої є *LLVM*-фронтенд. *Emscripten* підтримує трансляцію виведення бібліотеки *SDL* через *canvas*, а також надає підтримку *OpenGL* через *WebGL*. [14]

Серед проектів, які вдалося успішно перезібрати для роботи в браузері, можна відзначити ядро *NetBSD*, інтерпретатори *CPython*, *Ruby* і *Lua*, *PDF*-переглядач *Poppler*, систему рендеринга *TrueType*-шрифтів *FreeType*, декодувальник відео *H.264*, гру *Doom*, 3D гру *BananaBread* засновану на рушії *Sauerbraten*.

## 2.4. Діаграми

Для того, щоб краще розібратися в процесі поведінки системи активно використовувалася *CASE*-система «*Rational Rose*» за допомогою неї були складені діаграми видів діяльності, діаграми взаємодії, діаграма станів, на основі яких відбувається подальша програмна реалізація системи.

Щоб описати послідовності діяльностей, які представляють собою деякий стан, в якому щось виконується, найкраще використовувати діаграми видів діяльності. Дані діаграми встановлюють основні правила послідовності дій, які необхідно дотримуватися в тій чи іншій ситуації. Діаграми видів діяльності широко використовуються в проектуванні. Однак вони також служать в якості відмінного методу зображення обчислень або технологічних процесів на рівні абстракції, який підходить для аналізу. Графи видів діяльності можна використовувати для демонстрації різних рівнів деталізації обчислень. Тобто, якщо варіанти використання ставлять перед системою мету, то діаграма діяльності показує послідовність дій, необхідних для її досягнення. Дії (*action*) – це елементарні кроки, які не передбачають подальшу декомпозицію.

Діяльність може містити входять і/або вихідні дуги діяльності, що показують потоки управління і потоки даних. Якщо потік з'єднує дві діяльності, він є потоком управління. Якщо потік закінчується об'єктом, він є потоком даних.

Діяльність виконується, тільки тоді, коли готові всі її «входи», після виконання діяльність передає управління і (або) дані на свої «виходи».

На рис 2.2 представлена діаграма ідентифікації точок зору



Рис. 2.2. Діаграма ідентифікації точок зору

Оскільки цільова аудиторія користувачів – це середній шкільний вік, тому контролювати навчальний процес можуть як вчителі, так і батьки. На рис. 2.3 представлена діаграма ієрархії точок зору. Така діаграма допомагає побачити, що краще розробити програмний продукт таким чином, щоб його можна було використовувати для тренування та перевірки знань як на уроках у школі, так і вдома.

Для наочного представлення архітектури додатку наведені діаграми класів двох модулів, а саме «Vocabulary» та «Grammar» (див. рис. 2.4 та 2.5).

Діаграма станів (рис. 2.6) демонструє основні стани програми від початку гри до отримання результату, а діаграма послідовності ілюструє в які проміжки часу яка дія відбувається (рис. 2.7).

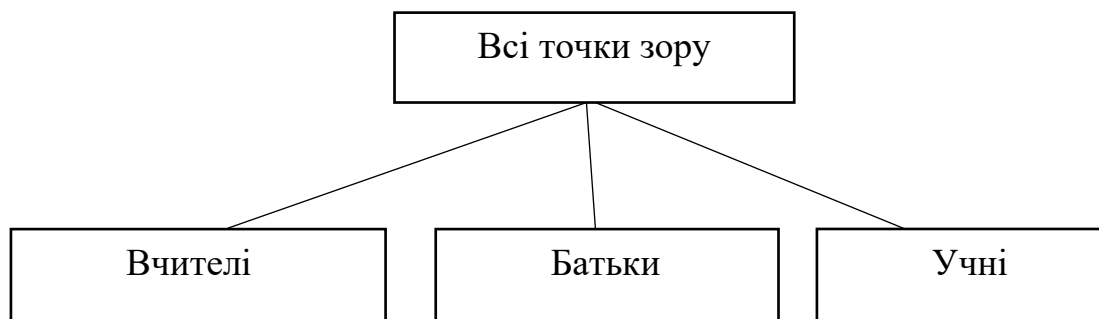


Рис. 2.3. Діаграма ієрархії точок зору

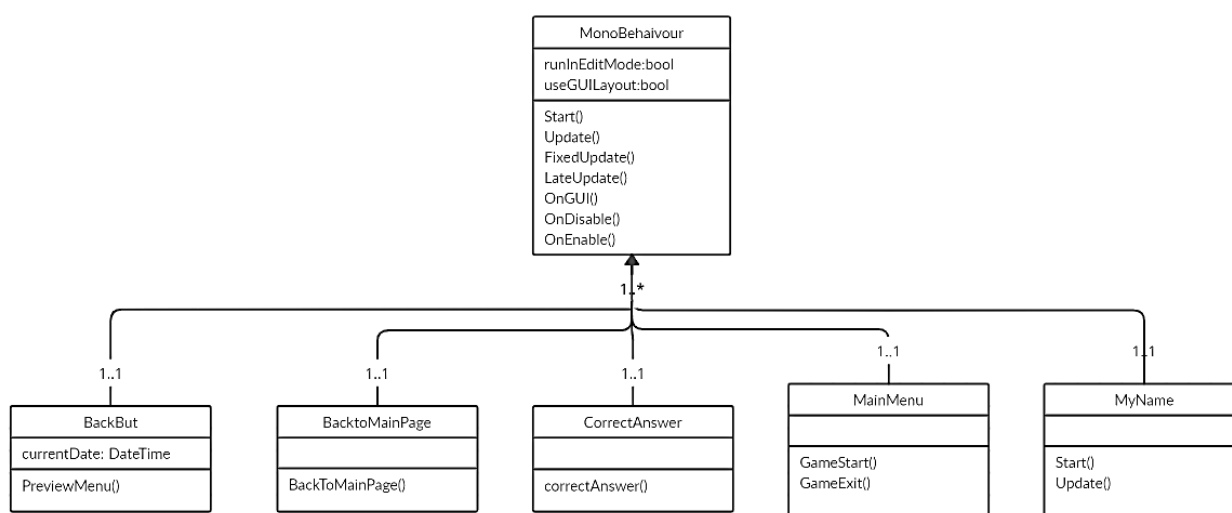


Рис. 2.4. Діаграма класів для модулю «Vocabulary»

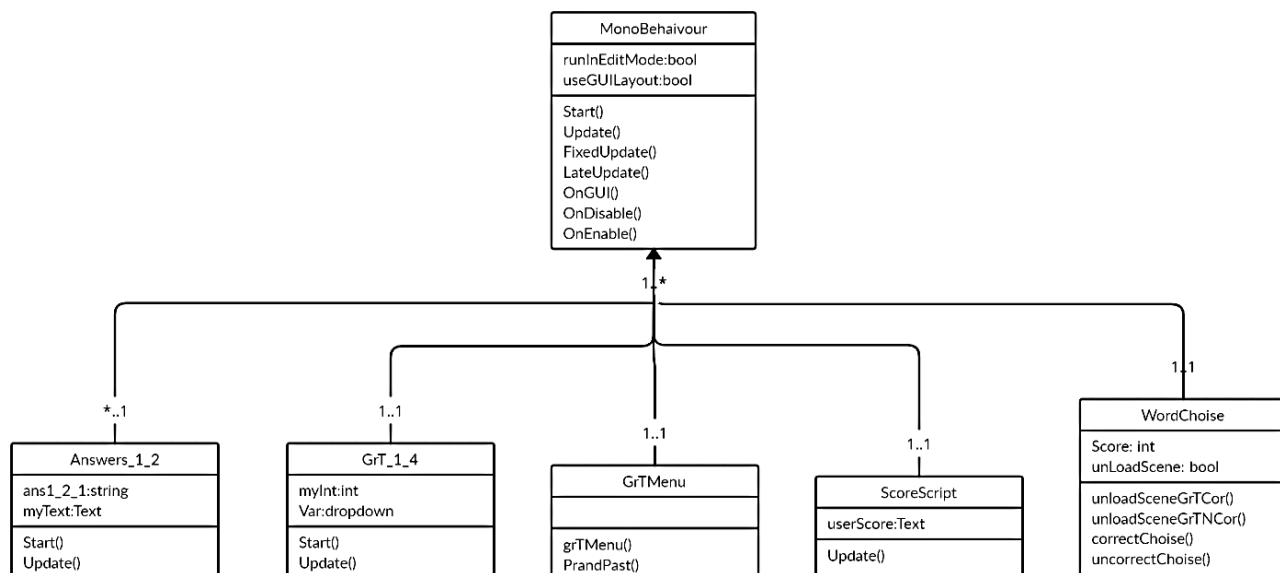


Рис. 2.5. Діаграма класів для модулю «Grammar»



Рис. 2.6. Діаграма станів

Діаграма варіантів використання (рис. 2.8) дозволяє побачити всі можливості програмного додатку, краще ознайомитися з його функціями, а також якомога якісніше покрити програмний додаток тестами.

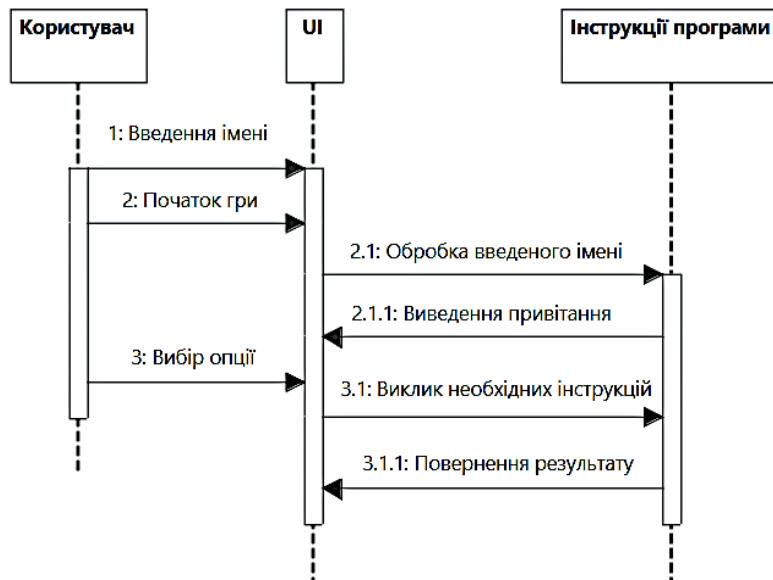


Рис. 2.7. Діаграма послідовності

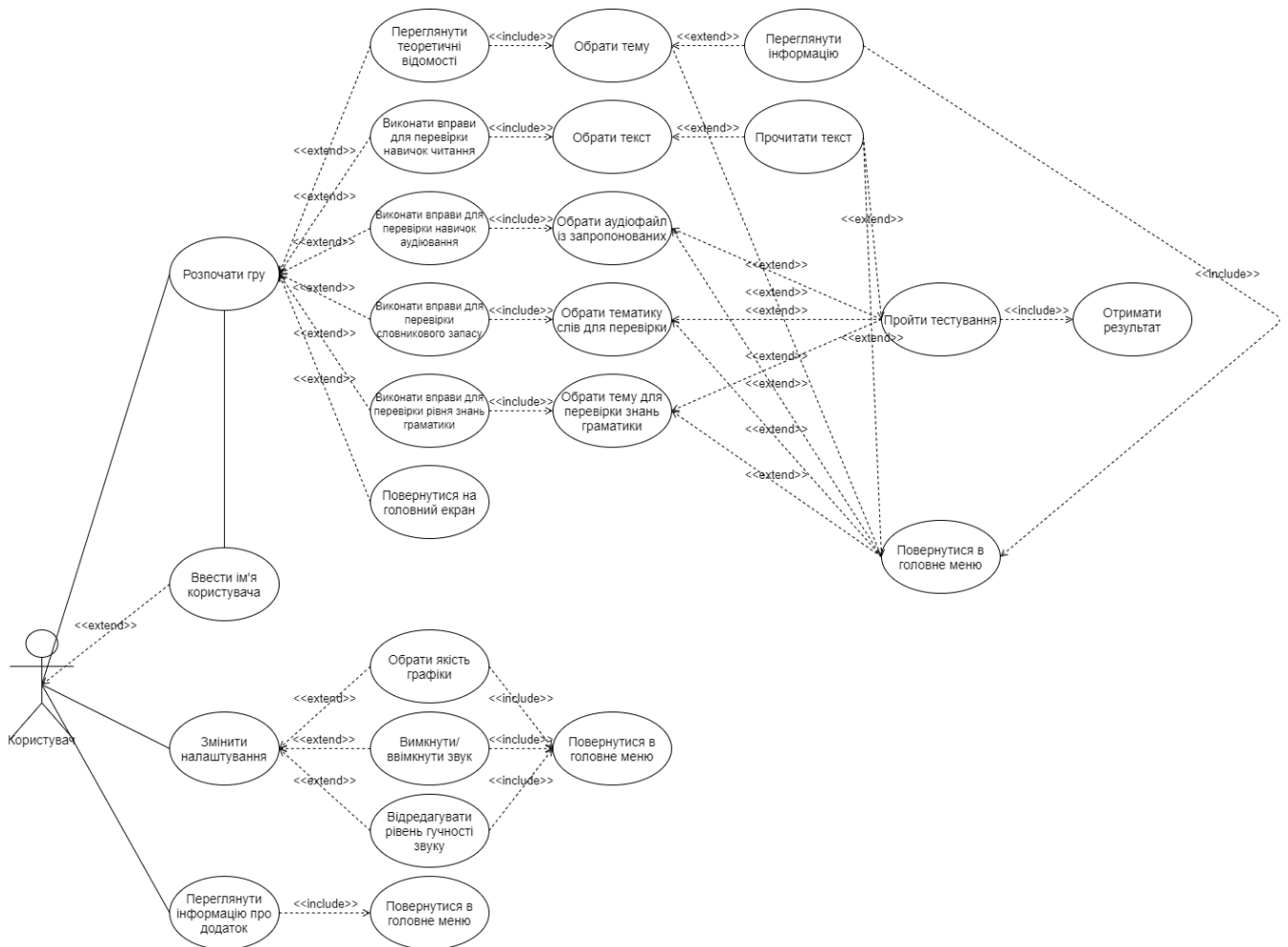


Рис. 2.8. Діаграма варіантів використання

Особливістю програми є те, що фон головного меню змінюється в залежності від часу доби відповідними зображеннями. Алгоритмічний опис такої функції представлено на рис. 2.9.

При проходженні вправ, користувач повинен знати чи відповів він правильно, чи зазнав поразки. Алгоритмічний опис функції, що представлено на рис. 2.10 ілюструє цей процес.

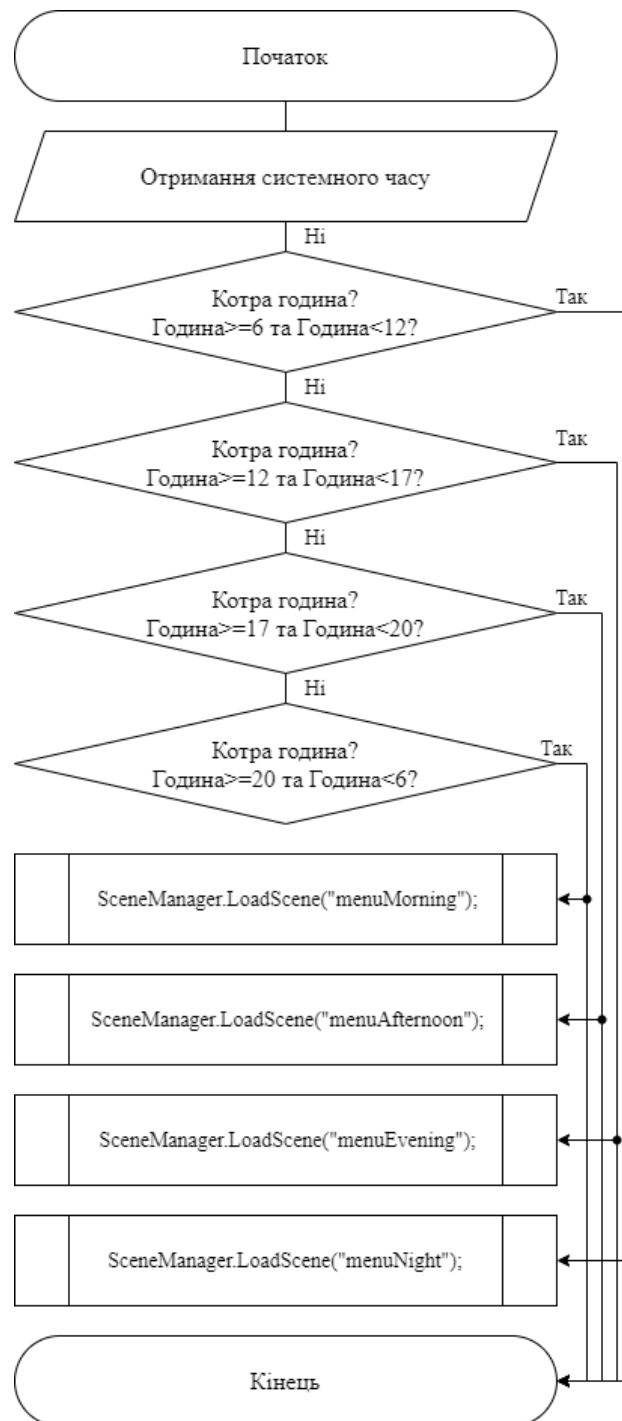


Рис. 2.9. Алгоритмічний опис функції зміни фону для головного меню

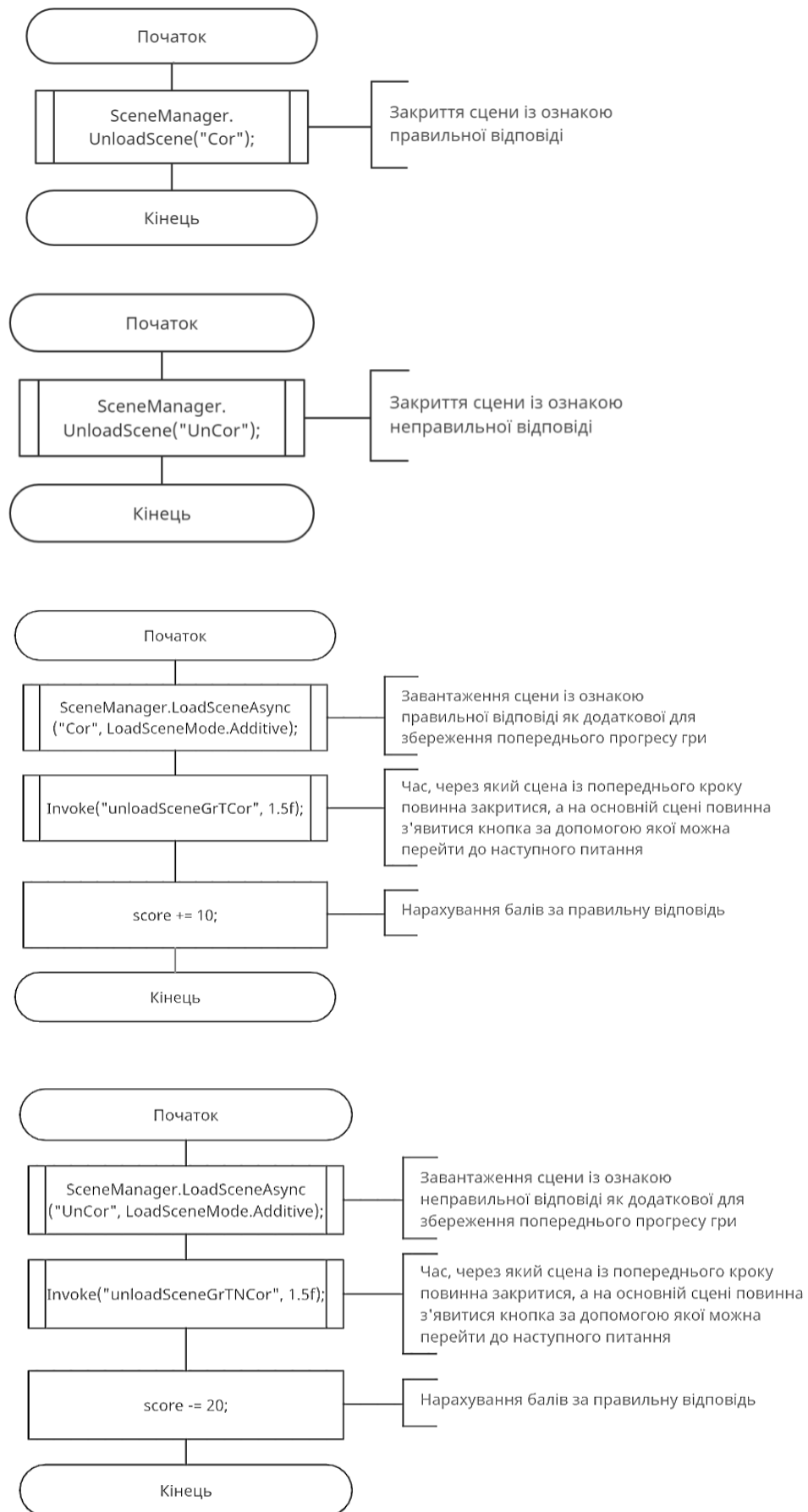


Рис. 2.10. Алгоритмічний опис модулів для класу *WordChoise*



## 2.5. Висновки до розділу

У даному розділі описані технології, що використовувалися для створення програмного модулю для вивчення іноземної мови на платформі *Unity*. Вибір таких програмних засобів обумовлений тим, що створюваний додаток повинен бути браузерним та вмикатися в залежності від того чи підтримує браузер бібліотеку *WebGL* чи ні, при тому що яка встановлена на комп'ютері операційна система значення не має. Це сприяє зручності використання та економії апаратних ресурсів комп'ютера.

РОЗДІЛ 3  
ПРОЄКТУВАННЯ ТА РОЗРОБКА ПРОГРАМНОГО МОДУЛЮ

3.1. Формування вимог до програмного модулю

У програмному модулі повинні бути реалізовані наступні можливості (функціональні вимоги):

- Наявність можливості відображати теоретичний матеріал;
- Наявність можливості проходити тестування;
- Наявність можливості переглянути результат проходження тестування;
- Сповіщення при невірній відповіді;
- Змінювати стандартний колір кнопки на червоний при невірній відповіді користувача;

- Наявність таймера;
- Наявність можливості керувати налаштуваннями.

Нефункціональні вимоги описують властивості та обмеження, що накладаються на програмний модуль. Для реалізації програмного модулю були виявлені наступні вимоги:

- Програмний модуль повинен бути реалізований на платформі *Unity* з використанням інтегрованого середовища *Visual Studio*;
- Програма повинна бути реалізована з використанням мови програмування *C#*;
- Програма повинна функціонувати у всіх версіях браузерів *Google Chrome* та *Microsoft Edge*.

Кафедра КСУ				НАУ 21 20 53 000 ПЗ			
<i>Виконав</i>	<i>Іванова М. С.</i>			Проектування та розробка програмного модулю	<i>Літера</i>	<i>Аркуш</i>	<i>Аркушів</i>
<i>Керівник</i>	<i>Апенько Н. В.</i>					49	61
<i>Консульт.</i>					СП-435 123		
<i>Норм. контр.</i>	<i>Тупота Є. В.</i>						
<i>Зав. Каф.</i>	<i>Литвиненко О.Є.</i>						

### 3.2. Опис програмного модуля

Навчальна програма пропонує користувачу декілька розділів з теоретичними відомостями для вивчення та перевіркою знань по кожному розділу.

Всього передбачено п'ять основних розділів: теоретичні відомості, розділ з текстами для читаннями та тестами для перевірки розуміння прочитаного, розділ з тестами для перевірки рівня словникового запасу та рівня знань граматики, розділ для перевірки навичок аудіювання.

Після проходження тестування користувач бачить результат, після чого може обрати чи пройти тестування знову, чи повернутися до головного меню та за бажанням обрати іншу опцію.

### 3.3. Написання коду

Приклад 1.

```
public class MainMenu : MonoBehaviour
{
    public void startGame()
    {
        var currentDate = System.DateTime.Now;
        if (currentDate.Hour >= 6 && currentDate.Hour < 12)
            SceneManager.LoadScene("menuMorning");
        else if (currentDate.Hour >= 12 && currentDate.Hour < 17)
            SceneManager.LoadScene("menuAfternoon");
        else if (currentDate.Hour >= 17 && currentDate.Hour <= 20)
            SceneManager.LoadScene("menuEvening");
        else if ((currentDate.Hour > 20 && currentDate.Hour <= 24) ||
(currentDate.Hour >= 0 && currentDate.Hour < 6))
    }
}
```

```

        SceneManager.LoadScene("menuNight");
    }
}

```

У наведеному вище прикладі програмного коду здійснюється перевірка системного часу, згідно з якою змінюється фон головного меню. Спочатку, і змінну *currentDate* ми записуємо системний час, потім виокремлюємо лише години, порівнюємо та завантажуюмо фон відповідно до часу доби.

Приклад 2.

```

public class WordChoise : MonoBehaviour
{
    public static int score = 0;
    public static bool unloadScene = false;
    public void unloadSceneGrTCor()
    {
        SceneManager.UnloadScene("Cor");
    }
    public void unloadSceneGrTNCor()
    {
        SceneManager.UnloadScene("UnCor");
    }
    public void correctChoise()
    {
        SceneManager.LoadSceneAsync("Cor", LoadSceneMode.Additive);
        Invoke("unloadSceneGrTCor", 1.5f);
        score += 10;
    }
    public void uncorrectChoise()
    {
        SceneManager.LoadSceneAsync("UnCor", LoadSceneMode.Additive);
        Invoke("unloadSceneGrTNCor", 1.5f);
    }
}

```

```
        score -= 20;
    }
}
```

У даному класі *WordChoise* маємо чотири методи: 2 методи, що ідентифікують правильну відповідь користувача, та 2 методи, що ідентифікують неправильну відповідь. У *Unity* кожне зображення на екрані називається сцена. Таким чином *public void correctChoise()* та *public void uncorrectChoise()* – методи правильної та неправильної відповідей відповідно. Після того, як користувач вибрав правильний на його думку варіант відповіді, спрацьовують та чи інша функція. Спочатку сцена з ознакою правильної/неправильної відповіді завантажується в додатковому режимі, щоб не втратився прогрес гри, на певну кількість часу: це реалізується за допомогою системної функції *Invoke()*, якій в параметри ми передаємо функцію, в якій описали, що сцена з ознакою відповіді повинна закритися та час через який це повинно відбутися. Після цього до рахунку гравця в разі правильної відповіді додається 10 балів, в інакшому випадку списується 20 балів.

Приклад 3.

```
public class ScoreScript : MonoBehaviour
{
    public static Text userScore;

    private void Update()
    {
        userScore = GetComponent<Text>();
        userScore.text = "Your score: " + WordChoise.score.ToString();
    }
}
```

У прикладі 3 виконується робота безпосередньо із ігровим рахунком користувача. Для відображення рахунку необхідно створити змінну типу *Text*, для того, щоб вона була доступна поза межами даного класу використовується

специфікатор доступу *public static*. Наступним кроком, створена змінна ідентифікується як об'єкт, в який записується фраза: «*Your score:*» та змінна *score* з класу *WordChoise*, яка конвертується в рядок. Останні два кроки виконуються в локальній функції *Update()*, яка при одноразовому виклиці постійно виконує інструкції, оновлюючи дані 1 раз секунду, до точки зупину.

#### 3.4. Висновки до розділу

При написанні даного розділу було спроектовано роботу програмного модулю для опанування іноземної мови на платформі *Unity*. При проєктуванні було створено діаграми взаємодії, станів, та класів, а також написаний код програмного додатку, який представлений у додатку Б пояснювальної записки.

Скрішоти роботи програми представлені у додатку А пояснювальної записки.

**РОЗДІЛ 4**  
**ТЕСТУВАННЯ ПРОГРАМНОГО МОДУЛЮ ТА ФОРМУВАННЯ**  
**ІНСТРУКЦІЙ КОРИСТУВАЧА**

**4.1. Функціональне тестування**

В рамках написання даного розділу проведено функціональне тестування програмного модулю.

Функціональне тестування – це тестування програмного забезпечення з метою перевірки відповідності реалізації функціональних вимог до технічного завдання, тобто можливість програмного забезпечення в певних умовах вирішувати задачі потрібні користувачам. Набір тестів для перевірки функціональності додатку представлений у таблиці 4.1.

Таблиця 4.1.

Результати функціонального тестування

№	Назва тесту	Вхідні дані	Очікуваний результат	Тест пройде-но?
1	Відображення головного екрану	Ввести ім'я	Введене ім'я зчитується коректно	Так
2	Відображення головного екрану	Натиснути на кнопку «About»	На екрані повинно з'явитися вікно інформації про програмний модуль	Так
3	Відображення головного екрану	Натиснути на кнопку «Exit»	Робота програмного модуля повинна припинитися	Так

<b>Кафедра КСУ</b>				<b>НАУ 21 20 53 000 ПЗ</b>			
<b>Виконав</b>	Іванова М. С.			Тестування програмного модулю та формування інструкцій користувача	<b>Літера</b>	<b>Аркуш</b>	<b>Аркушів</b>
<b>Керівник</b>	Апенько Н. В.					54	61
<b>Консульт.</b>					СП-435 123		
<b>Норм. контр.</b>	Тупота Є. В.						
<b>Зав. Каф.</b>	Литвиненко О.Є.						

№	Назва тесту	Вхідні дані	Очікуваний результат	Тест пройдено?
4	Відображення головного екрану	Натиснути на кнопку « <i>Play</i> »	Повинен відбутися перехід до меню опцій	Так
5	Відображення меню опцій	Натиснути на кнопку « <i>Grammar Reference</i> »	Повинен відбутися перехід до розділу із теоретичними відомостями	Так
6	Відображення меню опцій	Натиснути на кнопку « <i>Vocabulary</i> »	Повинен відбутися перехід до тестів на перевірку словникового запасу	Так
7	Відображення меню опцій	Натиснути на кнопку « <i>Reading Tests</i> »	Повинен відбутися перехід до розділу з текстами та тестами для перевірки прочитаного матеріалу	Так
8	Відображення меню опцій	Натиснути на кнопку « <i>Grammar Tests</i> »	Повинен відбутися перехід до розділу із тестам на граматику	Так
9	Відображення меню опцій	Натиснути на кнопку « <i>Listening Tests</i> »	Повинен відбутися перехід до розділу із тестами для перевірки навичок аудіювання	Так
10	Відображення меню опцій	Натиснути на кнопку « <i>Back</i> »	Повинен відбутися перехід до головного меню	Так
11	Вибір правильної відповіді	Натиснути на варіант відповіді	При виборі правильного варіанту на екрані з'явиться напис « <i>Correct!</i> »	Так



## 4.2. Формування інструкцій для користувача

Програма працює в режимі онлайн. Даний програмний модуль можна вмикати програму потрібно в одному із таких браузерів:

- *Google Chrome* (починаючи з версії 9);
- *Mozilla Firefox* (починаючи з версії 4.0);
- *Safari* (починаючи з версії 8.0);
- *Internet Explorer* (починаючи з версії 11);
- *Microsoft Edge*;
- *Opera* (починаючи з версії 12).

На першому екрані в поле для імені можна вводити будь-які символи, адже програма передбачає використання не тільки справжнього імені користувача, а також його нікнейму.

На головному екрані програми можна ввести ім'я, подивитися інформацію про програму, змінити налаштування та розпочати гру. Після того, як користувач розпочинає гру, відкривається головне меню програми, в якому представлено 5 режимів програми: теоретичні відомості, вправи на граматику, читання, аудіювання та перевірку словникового запасу.

Після того, як користувач обирає один із режимів, він переходить у меню вибору теми. Після вибору теми, користувач проходить тестування, якщо це ігрові режими та знайомиться з граматикою іноземної мови, якщо це режим теоретичних відомостей. Після проходження тестування користувач бачить свій результат на екрані.

На будь-якому із етапів, крім ігрових режимів можна повернутися в головне меню або вийти із програми, натомість в ігрових режимах після початку тестування можливості покинути гру не передбачено.

### 4.3. Висновки до розділу

В даному розділі проводилося тестування написаного програмного модулю, що дозволило перевірити правильність його роботи, побачити слабкі та сильні сторони.

Також були складені інструкції користувача та обумовлені необхідні програмні характеристики для використання даного програмного модулю.

## ВИСНОВКИ

Метою даного дипломного проекту є розробка програмного модулю для вивчення іноземної на платформі *Unity*.

Для досягнення цієї мети були поставлені наступні завдання: проєктування діаграм ідентифікації точок зору; проєктування *UML*-діаграм; розробка програмного модулю; тестування готового програмного продукту; написання інструкцій користувача.

Вивчення ІМ наразі є актуальним питанням для більшості людей. Вивчати ІМ можна різними способами: із репетитором, на курсах, за допомогою підручників, дивлячись фільми мовою оригіналу, деякі навіть їдуть до країни, мову якої бажають вивчити, адже мовне середовище сприяє кращому та швидшому засвоєнню. Проте, навчання потребує систематичності і тому із розвитком ІТ з'явилися ПЗ, що допомагають регулярно поповнювати словниковий запас, вивчати граматику, розвивати навички читання, письма та аудіювання.

Перевага ПЗ перед підручниками у тому, що, по-перше, після виконання завдання одразу можна дізнатися результат засвоєння матеріалу, по-друге, вивчення мови відбувається в ігровій формі, що сприяє заохоченню не тільки дорослих, а також дітей та підлітків до НП, по-третє враховуючи різноманіття ПЗ, можна не виходячи із дому, дистанційно, знайти носія мови та спілкуватися із ним у форматі онлайн – переписуватися чи за допомогою відеозв'язку.

*Unity* – це крос-платформна ігрова платформа, розроблена *Unity Technologies*, вперше випущена у червні 2005 року на всесвітній конференції розробників *Apple Inc.* як ексклюзивна ігрова платформа для *Mac OS X*. З тих пір платформа поступово розширюється для підтримки різноманітних настільних, мобільних, консольних та віртуальних додатків.

Поєднуючи необхідність створення додатку, що допоможе при вивченні іноземної мови дітям середнього шкільного віку та потужну сучасну платформу

для створення програм *Unity* вдалося створити програмний продукт, що сприяє поширенню форми *Edutainment* у шкільному процесі.

В дипломному проєкті було представлено порівняльний аналіз готових програмних продуктів для вивчення іноземної мови. Здебільшого всі вони працюють лише локально на комп'ютері та їх використання залежить від системних характеристик комп'ютера.

Проаналізувавши результати першого розділу розглядалися платформи та технології, які б дозволили створити програмний продукт, який буде працювати онлайн.

В третьому розділі було описано практичну реалізацію системи: принципи роботи основних модулів, характеристики програми та сформовані вимоги до програмного продукту.

В ході виконання дипломного проєкту було розроблено програмний модуль опанування іноземної мови на платформі *Unity* який працює онлайн.

Підводячи підсумки дипломного проєкту, можна зробити висновок, що поставлена мета дослідження та розробки була досягнута. Тепер, завдяки створеному додатку вчителі в школах можуть урізноманітнити навчальний процес інтерактивною грою, а батьки долучитися до інтелектуального розвитку своєї дитини.

## СПИСОК БІБЛОГРАФІЧНИХ ПОСИЛАНЬ ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Іванова М. С. *Edutainment* як спосіб поліпшення вивчення іноземної мови / Марина Сергіївна Іванова. // Національний авіаційний університет "Політ. Сучасні проблеми науки". – 2021. – С. 42.

2. Можарова О. А. Использование компьютерных программ в обучении иностранным языкам [Электронный ресурс] / О. А. Можарова // Студенческая научная сессия - 2020. – 2020. – Режим доступа до ресурсу: <https://www.rgph.vsu.ru/ru/science/sss/>.

3. Вагнер М. Л. Использование компьютерных программ в изучении английского языка в вузе / Моника-Наталия Лауренсовна Вагнер. // Интернет-журнал «Мир науки». – 2017. – №6. – С. 4.

4. Вишнякова Е. Д. Использование компьютерных технологий на уроках английского языка в средней школе [Электронный ресурс] / Е. Д. Вишнякова // Оренбургский государственный университет. – 2017. – Режим доступа до ресурсу: <http://elib.osu.ru/handle/123456789/5614>.

5. Войтко С. А. Об использовании информационно-коммуникационных технологий на уроках английского языка [Электронный ресурс] / Сергей Александрович Войтко // Интернет-журнал Фестиваль педагогических идей «Открытый урок». – 2007. – Режим доступа до ресурсу: <https://urok.1sept.ru/articles/415914>.

6. Альбрехт К. Н. Компьютер как эффективное средство обучения иностранному языку [Электронный ресурс] / Ксения Николаевна Альбрехт // Электронный научный журнал "Информационно-коммуникационные технологии в педагогическом образовании". – 2011. – Режим доступа до ресурсу: <http://infed.ru/articles/63/>.

7. *Rosetta Stone* — програма для вивчення іноземних мов [Електронний ресурс] – Режим доступа до ресурсу: <http://frenglish.ru/10-eng-learn-courses-rosetta.html>.

8. *Tell Me More Performance* — програма для вивчення іноземних мов [Електронний ресурс] – Режим доступу до ресурсу: <http://frenglish.ru/10-eng-learn-courses-rosetta.html>.

9. *Bridge to English* — програма для вивчення іноземної мови [Електронний ресурс] – Режим доступу до ресурсу: [http://frenglish.ru/bridge\\_to\\_english.html](http://frenglish.ru/bridge_to_english.html).

10. 8 найкращих додатків для вивчення англійської мови [Електронний ресурс] – Режим доступу до ресурсу: <https://www.fluentu.com/blog/english-rus>

11. *Player Settings WebGL* [Електронний ресурс]. – 2021. – Режим доступу до ресурсу: <https://docs.unity3d.com/Manual/class-PlayerSettingsWebGL.html>.

12. *TroubleShootingEditor* [Електронний ресурс]. – 2021. – Режим доступу до ресурсу: <https://docs.unity3d.com/2019.4/Documentation/Manual/TroubleShootingEditor.html>.

13. *WebGL Browser Compatibility* [Електронний ресурс]. – 2021. – Режим доступу до ресурсу: <https://docs.unity3d.com/Manual/webgl-browsercompatibility.html>.

14. *WebGL graphics* [Електронний ресурс]. – 2021. – Режим доступу до ресурсу: <https://docs.unity3d.com/Manual/webgl-graphics.html>.

15. *WebGL-templates* [Електронний ресурс]. – 2021. – Режим доступу до ресурсу: <https://docs.unity3d.com/Manual/webgl-templates.html>.

16. Бойченко С.В., Іванченко О.В. Положення про дипломні роботи (проекти) випускників Національного авіаційного університету. – К.: НАУ, 2017. – 63 с.

17. ДСТУ 3008-95 Документація. Звіти у сфері науки і техніки. Структура і правила оформлення.

## ДОДАТОК А

Знімки екрану при роботі програми



Рис. А.1. Головний екран

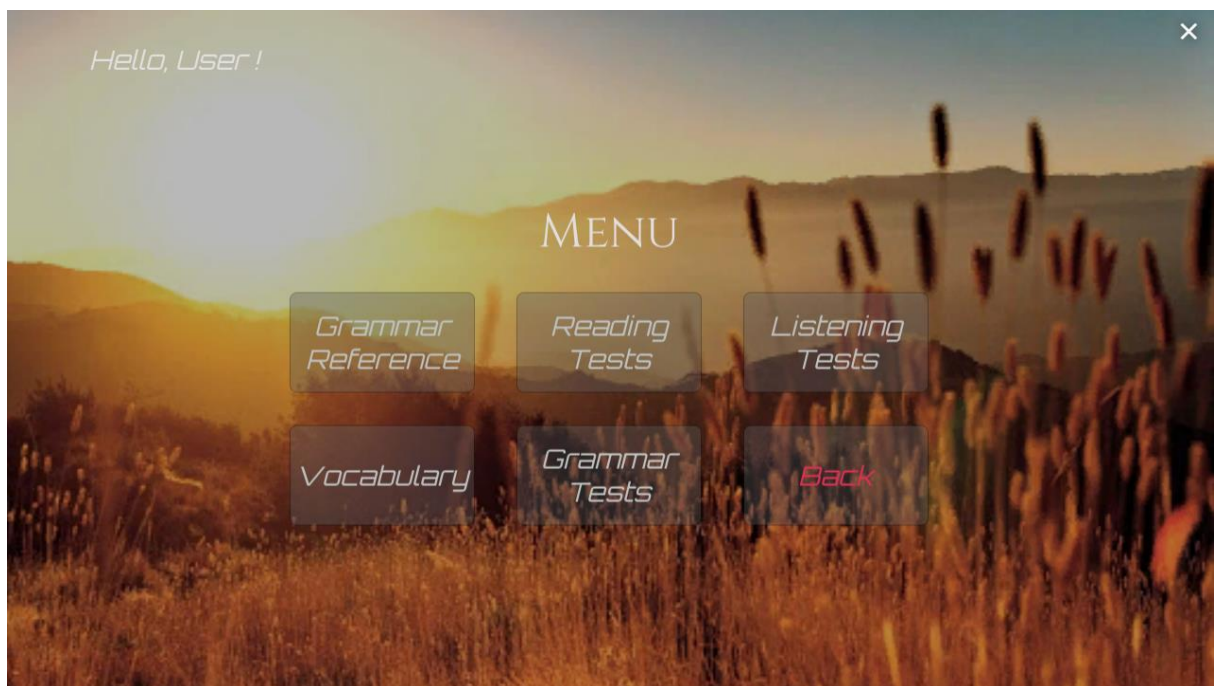


Рис. А.2. Меню програми (ранок)



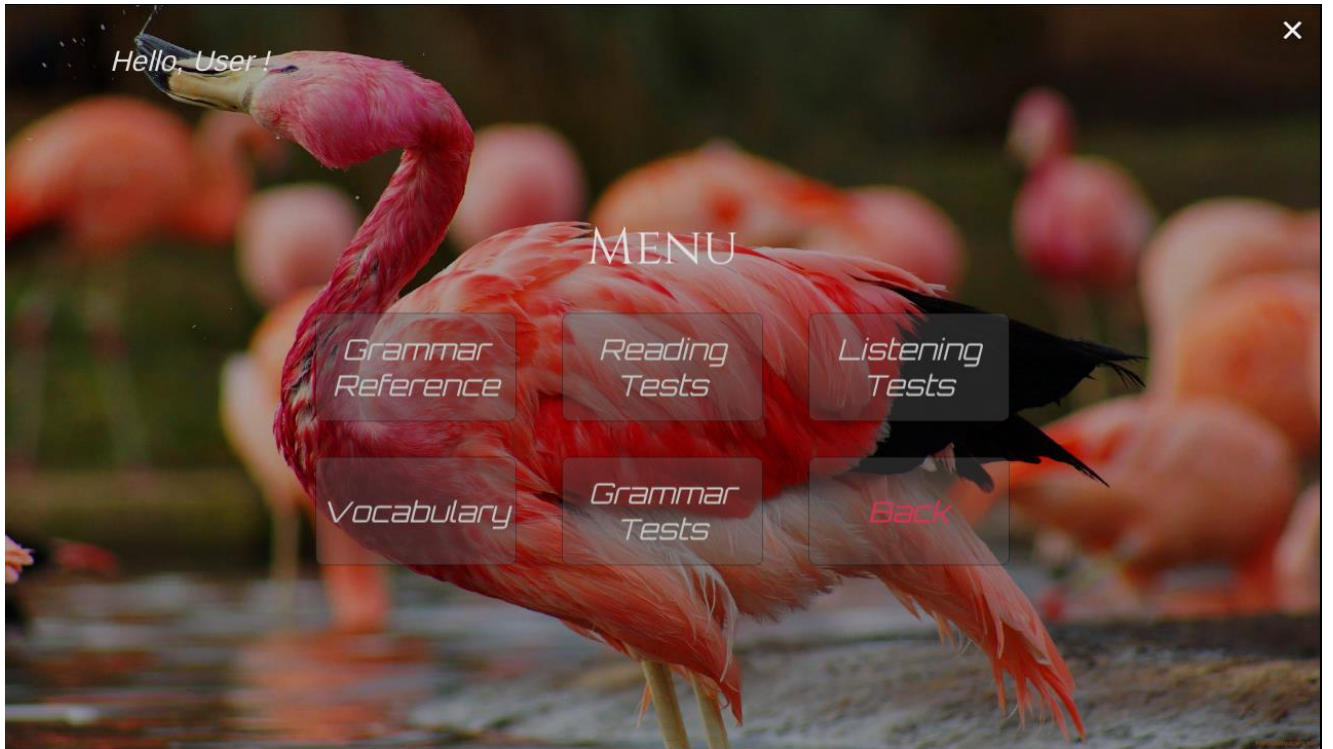


Рис. А.3. Меню програми (день)

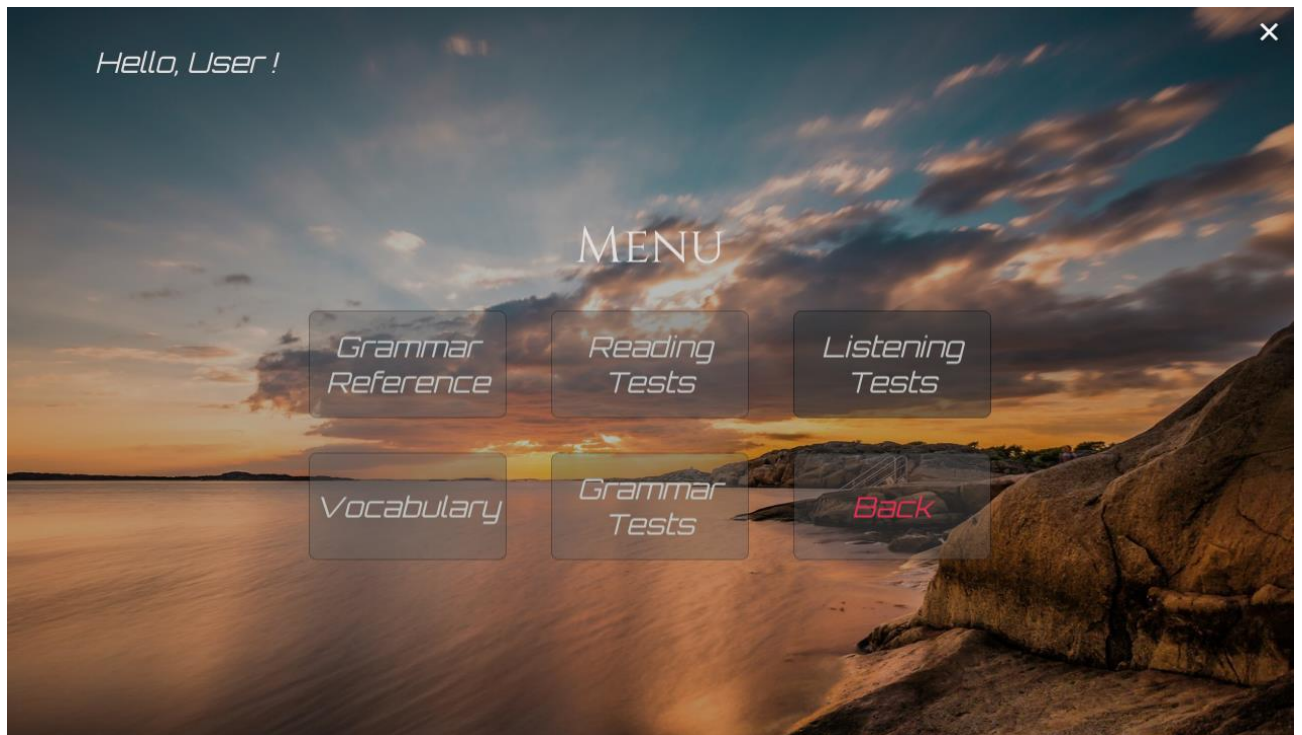


Рис. А.4. Меню програми (вечір)



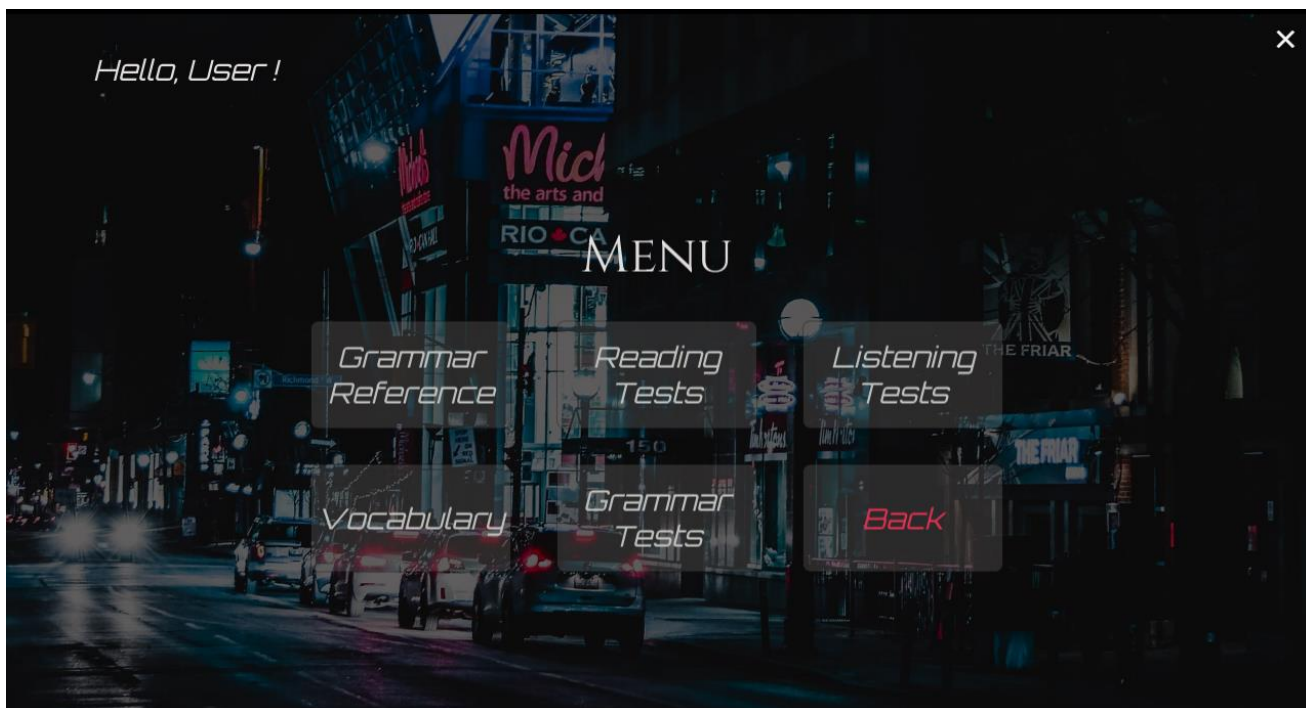


Рис. А.5. Меню програми (ніч)

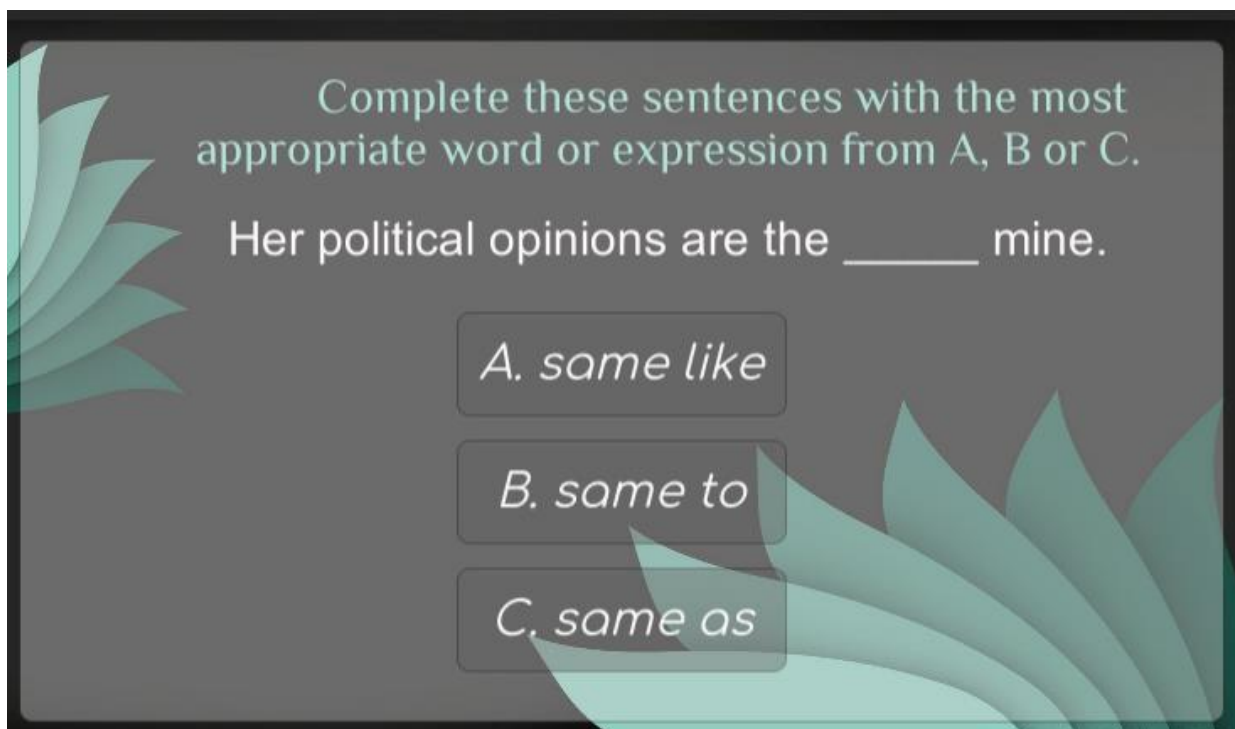


Рис. А.6. Розділ «Vocabulary»

**Unit 1** 1.1 What's happening in the pictures?  
Choose from these verbs:



crossing   hiding   scratching   taking   tying   waving



She's \_\_\_\_\_ a picture.

Your score: 0

Next

Рис. А.6. Розділ «*Grammar*»

Correct !

Рис. А.8. Напис, що з'являється при натисканні на правильний варіант відповіді

## ДОДАТОК Б

### Програмний код розробленого програмного модуля

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;
public class MainMenu : MonoBehaviour
{
    public void startGame()
    {
        var currentDate = System.DateTime.Now;
        if (currentDate.Hour >= 6 && currentDate.Hour < 12)
            SceneManager.LoadScene("menuMorning");
        else if (currentDate.Hour >= 12 && currentDate.Hour < 17)
            SceneManager.LoadScene("menuAfternoon");
        else if (currentDate.Hour >= 17 && currentDate.Hour <= 20)
            SceneManager.LoadScene("menuEvening");
        else if ((currentDate.Hour > 20 && currentDate.Hour <= 24) //
(currentDate.Hour >= 0 && currentDate.Hour < 6))
            SceneManager.LoadScene("menuNight");
        //SceneManager.LoadScene("1");
    }
}
public class BacktoMainPage : MonoBehaviour
{
    public void BackToMainPage()
    {
        SceneManager.LoadScene("SampleScene");
    }
}
```

```

    }
}
public class MyName : MonoBehaviour
{
    public static string myName;
    private Text myText;
    void Start()
    {
        myText = GetComponent<Text>();
    }
    void Update()
    {
        myName = myText.text.ToString();
    }
}
public class Time : MonoBehaviour //format24
{
    public Text timeText;
    void Update()
    {
        var currentDate = System.DateTime.Now;
        float hours = currentDate.Hour;
        float minutes = currentDate.Minute;
        float seconds = currentDate.Second;
        timeText.text = string.Format("{0:00}:{1:00}:{2:00}", hours, minutes,
seconds);
    }
}
public class Answers_1_2 : MonoBehaviour
{

```

```

public static string ans1_2_1;
private Text myText;
void Start()
{
    myText = GetComponent<Text>();
    myText.GetComponent<Text>().color = new Color(50/255.0f, 50 / 255.0f,
50 / 255.0f, 250 / 255.0f);
}
void Update()
{
    ans1_2_1 = myText.text.ToString();
}
public void OnEndEdit(/*string answer, char A, char B*/)
{
    if(ans1_2_1 == "f" || ans1_2_1 == "F")
    {
        WordChoise.score += 10;
        myText.GetComponent<Text>().color = new Color(80 / 255.0f, 241 /
255.0f, 120 / 255.0f, 100 / 255.0f);
    }
    else
    {
        WordChoise.score -= 20;
        myText.GetComponent<Text>().color = new Color(541 / 255.0f, 80 /
255.0f, 98 / 255.0f, 100 / 255.0f);
    }
}
}
public class GrTMenu : MonoBehaviour
{

```

```

public void grTMenu()
{
    SceneManager.LoadScene("GrTMenu");
}
public void PrandPast()
{
    SceneManager.LoadScene("PrPastMenu");
}
}
public class ScoreScript : MonoBehaviour
{
    private static Text userScore;

    private void Update()
    {
        userScore = GetComponent<Text>();
        userScore.text = "Your score: " + WordChoise.score.ToString();
    }
}
public class WordChoise : MonoBehaviour
{
    public static int score = 0;
    public static bool unloadScene = false;
    public void unloadSceneGrTCor()
    {
        SceneManager.UnloadScene("Cor");
    }

    public void unloadSceneGrTNCor()
    {

```

```

        SceneManager.UnloadScene("UnCor");
    }
    public void correctChoise()
    {
        SceneManager.LoadSceneAsync("Cor", LoadSceneMode.Additive);
        Invoke("unloadSceneGrTCor", 1.5f);
        score += 10;
    }
    public void uncorrectChoise()
    {
        SceneManager.LoadSceneAsync("UnCor", LoadSceneMode.Additive);
        Invoke("unloadSceneGrTNCor", 1.5f);
        score -= 20;
    }
}
public class ReadingNext : MonoBehaviour
{
    public void ReadingMenu()
    {
        SceneManager.LoadScene("ReadingMenu");
    }
    public void Text_1_1_Text()
    {
        SceneManager.LoadScene("1_1_Text");
    }
    public void Text1_1_Next()
    {
        SceneManager.LoadScene("1_1_Q");
    }
}
}

```

```

public class CorrectAnswer : MonoBehaviour
{
    public void unloadSceneCA()
    {
        SceneManager.UnloadScene("PracticeCorrect");
    }
    public void correctAnswer()
    {
        SceneManager.LoadSceneAsync("PracticeCorrect",
LoadSceneMode.Additive);
        //SceneManager.LoadScene("PracticeCorrect");
        Invoke("unloadSceneCA", 1.5f);
    }
}
public class UncorrectAnswer : MonoBehaviour
{
    public void unloadSceneUA()
    {
        SceneManager.UnloadScene("PracticeUncorrect");
    }
    public void uncorrectAnswer()
    {
        //SceneManager.LoadScene("PracticeUncorrect");
        SceneManager.LoadSceneAsync("PracticeUncorrect",
LoadSceneMode.Additive);
        Invoke("unloadSceneUA", 1.5f);
    }
}
public class VocButton : MonoBehaviour
{

```



```
public void vocButton()  
{  
    SceneManager.LoadScene("VocMenu");  
}  
}
```